Preprint typeset in JHEP style - HYPER VERSION

LU-TP 08-20
MCnet/08/13



# Extending CKKW-merging to One-Loop Matrix Elements*

**Nils Lavesson and Leif Lönnblad**

*Dept. of Theoretical Physics, Sölvegatan 14A, S-223 62 Lund, Sweden*
*E-mail:* `Nils.Lavesson@thep.lu.se` *and* `Leif.Lonnblad@thep.lu.se`

Abstract: We extend earlier schemes for merging tree-level matrix elements with parton showers to include also merging with one-loop matrix elements. In this paper we make a first study on how to include one-loop corrections, not only for events with a given jet multiplicity, but simultaneously for several different jet multiplicities. Results are presented for the simplest non-trivial case of hadronic events at LEP as a proof-of-concept.

Keywords: QCD, Jets, Parton Model, Phenomenological Models.

*Work supported in part by the Marie Curie RTN "MCnet" (contract number MRTN-CT-2006-035606).

## Contents



## 1. Introduction

One of the big theoretical challanges with LHC physics is the description of states with many hard jets. The high energy and large rapidity range for jets at the LHC means that producing multi-jet events from QCD processes is more likely than ever before. These states compose the background for many of the channels that could contain new physics. It is therefore important to get as good description as possible for the multi-jet states within QCD.

Monte Carlo event generators have become standard tools for simulating events in a particle collider. These event generators try to simulate events with all the characteristics of a real event. The theoretical basis is a parton shower, which is combinded with hadronization models to produce the final-state hadrons. The hadronization models are phenomenological models that only work reliably when all the partons in the soft and collinear limit have been simulated corretly in the shower. Parton showers are based on expanding the emission probabilities in this limit, which makes them suitable to use together with the hadronization models. Although parton showers have been used to describe a wide range of results with good accuracy, it is well known that they cannot give a good description of observables sensitive to emissions away from the collinear and soft regions.

The way to improve the description of multi-jet states is to include exact matrix elements. The matrix elements describe these states well, but do not provide a way of describing emissions in the soft or collinear limit. In fact, the matrix elements are divergent in these limits and have to be regulated using a cutoff. To correctly describe the final-state hadrons in multi-jet events, the matrix elements and the parton shower descriptions need to be combined. This has been done for tree-level matrix elements using algorithms such as CKKW [1,2], CKKW-L [3,4], MLM [5,6] and Pseudo-Shower [7].

In recent years a lot of effort has been put into calculating one-loop matrix elements to be able to predict observables to next-to-leading order (NLO) accuracy. There are several program available to do this, *e.g.* MCFM [8] and NLOJET++ [9]. Currently efforts are being made to automate the whole procedure and make more processes available, including significantly higher parton muliplicities (MCFM and BlackHat [10]). The one-loop matrix element calculations contain important QCD contributions which cannot be simulated with tree-level matrix elements nor with parton showers. Preferably both the tree-level and one-loop matrix elements as well as the parton shower should be used consistently together. Lacking such a complete description, the uncertainties in a given NLO calculation due to parton showers and hadronization are typically estimated using a separate approximate Monte Carlo simulation. Alternatively, for certain observables, it is possible to combine a NLO calculation with analytically resummed parton-shower correction together with a semi-universal power correction giving the hadronization correction. However, it would clearly be advantegous if the one-loop matrix elements could be used together with parton showers and hadronization models in a more consistent manner.

A few algorithms have been presented to merge one-loop matrix elements with parton showers. The two main algorithms are MC@NLO [11,12] and POWHEG [13,14], but they are limited to only including the one-loop matrix element for the lowest order process. A similar algorihtm for $e^+e^-$ to three jets was presented in [15–17]. Other groups have made proposals on how to go beyond this and include one-loop matrix elements also for higher multiplicities [18–21][1], but none have so far presented a complete implementation.

In this paper we present a general algorithm to include one-loop matrix elements of any order in the strong coupling together with parton showers. The idea is to take the first two terms in order of $\alpha_{\mathrm{s}}$ from the one-loop matrix element and all higher order terms from the parton shower. Two different events samples are generated. The first sample consists of events which are generated according to one-loop matrix elements and dressed using a parton shower. The second sample are events generated with a parton shower corrected with tree-level matrix elements, where the first two terms in the $\alpha_{\mathrm{s}}$-expansion has been subtracted. The procedure is applied to all the different multiplicities one wishes to calculate and in the end all the samples are added.

Calculating the first two terms in orders of $\alpha_{\mathrm{s}}$ in the shower introduces complications because they include the first term in an expansion of the Sudakov form factor. In addition the running $\alpha_{\mathrm{s}}$ used in the shower also gives a contribution to the terms at this order. These complications need to be dealt with in order to have a consistent algorithm.

[1] In particular [18] is very close in spirit to the strategy presented here.

Our method uses the same philosophy as CKKW-L, which means that phase space is split up in two different regions using a merging scale, and the corrections to the matrix elements are simulated using the shower. However, it should be noted that most of what is presented here could also be used together with the CKKW algorithm, where the corrections are calculated analytically. Using the shower rather than doing analytical calculation means that non-leading terms, such as energy–momentum conservation and recoil treatments, included in the shower, are also included in the corrections to the matrix elements.

Although it would be more interesting to simulate jets at the LHC, we limit ourselves to LEP physics in this paper. The reason is that the inclusion of parton densities causes a number of additional complications that needs to be studied further before an algorithm valid also for incoming hadrons can be presented.

The outline of this paper is the following. In section 2 some key concepts of parton shower and matrix elments are reviewed, which are then used in the description of the merging. Section 3 describes the various steps in the algorithm and how to calculate all the terms needed. The algorithm is implemented toghether with ARIADNE [22] and the results are presented in section 4. Finally in section 5 our conclusions are presented.

## 2. Theory

This section contains some basic properties of parton showers and matrix elements, including both tree-level and one-loop matrix elements. After the theoretical background has been established our algorithm for merging one-loop matrix elements and parton shower is presented.

### 2.1 Parton showers

Emissions in the soft and collinear limit can be resummed to all orders using a parton shower. This is done using only the dominant behavior in this limit which is translated into an emission probablility. The other main component is the assumption that the emissions can be ordered. To make the results exclusive, the probability for emitting a parton also includes the probability that no emission has occured at a higher scale, which is known as a Sudakov form factor. The result is a formalism where each emission can be considered individually and generated according to a resonably simple probability distribution, which is ideal for computer simulations.

The parton shower is a good approximation for emissions near the soft and collinear limits and can give a nice description for a large range of observables, since the approximation is valid for the bulk of the cross section. There are important exceptions though, which occur mainly when you have several hard partons emitted at wide angle.

Different shower models use different choices to specify the ordering variable, the most common choices being transverse momentum, angle and virtuality. To formulate the parton cross sections in a general way we simply denote the ordering variable $\rho$. The emission probability is the product of a splitting function, which is a function of the emissions kinematics, and the stong coupling $\alpha_{\mathrm{s}}$. Most showers use a running $\alpha_{\mathrm{s}}$ with a transverse momentum as the scale, which is not necessarily equal to the ordering variable. However, here we assume that the ordering variable is the same as the scale in $\alpha_{\rm s}$ for notiational convenience. The cross sections for the parton multiplicities for a shower that has evolved down to the shower cutoff ($\rho_c$) can be written in the following way.

$$\begin{aligned}
d\sigma_0 &= C_0^{\rm PS}(\Omega_0^{\rm PS})\, \Delta_{S_0}(\rho_0,\rho_c) d\Omega_0^{\rm PS} \\
d\sigma_1 &= C_1^{\rm PS}(\Omega_1^{\rm PS})\, \alpha_{\rm s}(\rho_1)\Delta_{S_0}(\rho_0,\rho_1)\Delta_{S_1}(\rho_1,\rho_c) d\Omega_1^{\rm PS} \\
d\sigma_2 &= C_2^{\rm PS}(\Omega_2^{\rm PS})\, \alpha_{\rm s}(\rho_1)\alpha_{\rm s}(\rho_2)\Delta_{S_0}(\rho_0,\rho_1)\Delta_{S_1}(\rho_1,\rho_2)\Delta_{S_2}(\rho_2,\rho_c) d\Omega_2^{\rm PS} \\
&\vdots \\
d\sigma_n &= C_n^{\rm PS}(\Omega_n^{\rm PS})\Delta_{S_n}(\rho_n,\rho_c) \prod_{i=1}^{n} \alpha_{\rm s}(\rho_i)\Delta_{S_{i-1}}(\rho_{i-1},\rho_i) d\Omega_n^{\rm PS} \\
&\vdots
\end{aligned} \tag{2.1}$$

The parton shower phase space is described by

$$\Omega_n^{\rm PS} = (\mathbf{q}_1,\ldots,\mathbf{q}_m,\rho_1,\vec{x}_1,\rho_2,\vec{x}_2,\ldots,\rho_n,\vec{x}_n), \tag{2.2}$$

where $\mathbf{q}$ denotes the momenta of the $m$ (usually two) outgoing partons at Born level, $\rho$ is the value of the ordering variable and $\vec{x}$ are the other kinematical variables that describe each emission. $\Delta_{S_n}(\rho_n,\rho_{n+1})$ denotes the Sudakov form factor which is the probability that no emission occurs from the $n$-parton state, $S_n$, between the scales $\rho_n$ and $\rho_{n+1}$, and the $C_n^{\rm PS}$-coefficients are the Born-level matrix element multiplied with the products of splitting functions in the shower and depend on all $\rho_i$ and $\vec{x}_i$ with $i \le n$.

The Sudakov form factors in the shower is an approximate way of calculating the virtual diagrams to all orders. In the angular ordered shower in HERWIG this is done by an analytical calculation based on the production scale of the various partons, but other showers, such as the parton showers in PYTHIA and the dipole shower in ARIADNE, uses the actual parton state when calculating the Sudakov form factor as an explicit no-emission probability. The algorithm described in later section requires Sudakov form factors that factorize ($\Delta_{S_i}(\rho_1,\rho_2)\Delta_{S_i}(\rho_2,\rho_3) = \Delta_{S_i}(\rho_1,\rho_3)$) which is the case if the Sudakov form factors only depend on the intermediate state. In this paper the notation used reflects the dependency on the parton state, which is denoted by a subscript. The Sudakov form factors can explicitly be written

$$\Delta_{S_n}(\rho_i,\rho_{i+1}) = \exp\left(-\int_{\rho_{i+1}}^{\rho_i} d\rho\,\alpha_{\rm s}(\rho)\Gamma_{S_n}(\rho)\right), \tag{2.3}$$

where $\Gamma$ denotes the branching probability for the specific parton state.

It should be noted that the way the virtual corrections are approximated in the shower means that the sum of all the parton cross sections is equal to the Born cross section:

$$\sum_{i=0}^{\infty} \sigma_i = \int C_0^{\rm PS}(\Omega_0^{\rm PS}) d\Omega_0^{\rm PS} = \sigma_{\rm Born}. \tag{2.4}$$

This means that the parton shower does not properly approximate the higher-order corrections to the total cross section. Instead it is common to include a $K$-factor by scaling all the cross sections with the N$^l$LO cross section divided by the Born cross section. However, it does nothing to improve the shape observables or the relative abundance of different parton multiplicities.

Measurements of $\alpha_{\mathrm{s}}$ have been done with better and better accuracy, typically using precision data from LEP (see *e.g.* [23]). If the same values of $\alpha_{\mathrm{s}}$ would be used within a parton shower it would not describe data well. The reason is that the shower has a tendency to underestimate emission probabilities, especially for hard emissions. The shower therefore gives a better description of data if a higher value of $\alpha_{\mathrm{s}}$ is used. To get the best possible fit, the parton shower implementations usually make $\alpha_{\mathrm{s}}$ tunable. This is frequently done by doing a one- or two-loop $\alpha_{\mathrm{s}}$ evolution and making $\Lambda_{\mathrm{QCD}}$ a parameter to be fit to data.

Making $\alpha_{\mathrm{s}}$ tunable is equal to modifying the scale used in the evaluation of $\alpha_{\mathrm{s}}$, apart from corrections related to flavour thresholds. This means that it is possible to use a different scale when evaluating $\alpha_{\mathrm{s}}$ and use the world average $\alpha_{\mathrm{s}}$ from precision measurements, which is illustrated in the equation below.

$$\alpha_{\mathrm{s}}^{\mathrm{PS}}(\rho) = \alpha_{\mathrm{s}}^{\mathrm{WA}}(b\rho) \tag{2.5}$$

This is something that is used in later sections of this paper.

It is possible to expand the parton cross sections in the shower in powers of $\alpha_{\mathrm{s}}$. To do this the exponential in each Sudakov form factors has to be expanded and the running of the coupling taken into account. The relative change of the cross section at each order of $\alpha_{\mathrm{s}}$ is denoted by $c_{n,m}$, where $n$ is the order in $\alpha_{\mathrm{s}}$ of the relevant tree diagram and $m$ is the number of loops. Using a renormalization scale, $\mu$, the parton cross sections in eq. (2.1) can be written as

$$\begin{aligned}
d\sigma_0 &= C_0^{\mathrm{PS}}(\Omega_0^{\mathrm{PS}})\,(1 + c_{0,1}^{\mathrm{PS}}(\Omega_0^{\mathrm{PS}})\alpha_{\mathrm{s}}(\mu) + c_{0,2}^{\mathrm{PS}}(\Omega_0^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}^2(\mu) + \ldots)d\Omega_0^{\mathrm{PS}}\\
d\sigma_1 &= C_1^{\mathrm{PS}}(\Omega_1^{\mathrm{PS}})\,\alpha_{\mathrm{s}}(\mu)(1 + c_{1,1}^{\mathrm{PS}}(\Omega_1^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}(\mu) + c_{1,2}^{\mathrm{PS}}(\Omega_1^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}^2(\mu) + \ldots)d\Omega_1^{\mathrm{PS}}\\
d\sigma_2 &= C_2^{\mathrm{PS}}(\Omega_2^{\mathrm{PS}})\,\alpha_{\mathrm{s}}^2(\mu)(1 + c_{2,1}^{\mathrm{PS}}(\Omega_2^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}(\mu) + c_{2,2}^{\mathrm{PS}}(\Omega_2^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}^2(\mu) + \ldots)d\Omega_2^{\mathrm{PS}}\\
&\;\vdots\\
d\sigma_n &= C_n^{\mathrm{PS}}(\Omega_2^{\mathrm{PS}})\alpha_{\mathrm{s}}^n(\mu)(1 + c_{n,1}^{\mathrm{PS}}(\Omega_n^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}(\mu) + c_{n,2}^{\mathrm{PS}}(\Omega_n^{\mathrm{PS}},\mu)\alpha_{\mathrm{s}}^2(\mu) + \ldots)d\Omega_n^{\mathrm{PS}}\\
&\;\vdots
\end{aligned} \tag{2.6}$$

All the higher order changes to the cross section (except $c_{0,1}$ in case the Born level contains no powers of $\alpha_{\mathrm{s}}$, which is assumed here) have a dependence on the renormalization scale, when one expands in terms of a fixed coupling constant. This happens because the shower uses a running $\alpha_{\mathrm{s}}$ and when changing the scale to $\mu$ there are residual terms that needs to be absorbed into the parton shower coefficients. These effects are described further in later sections.

Note that the parton shower provides a value for the parton cross sections that includes terms of all orders in $\alpha_{\mathrm{s}}$, but the coefficients are only approximately correct. The goal of merging algorithms is to replace some of the coefficients by the exact results in order to minimize the effects of the approximations done in the parton shower. In the following we discuss how these terms can be calculated and what kind of results one can achieve.

### 2.2 Matrix elements

Processes calculated through matrix elements means that one is calculating the amplitudes of the Feynman diagrams directly. This is easy to do for $2 \to 2$ processes, but gets increasingly difficult for larger number of external legs or if one includes loops.

To calculate an observable using matrix elements is equivalent to exactly calculating the terms in an $\alpha_{\mathrm{s}}$ expansion one term at a time. The advantage of matrix elements is that they are exact up to the calculated order. In certain regions of phase space this approach works fine, but for collinear and soft emissions there are divergencies in the matrix elements which prevents the $\alpha_{\mathrm{s}}$ expansion from converging.

The leading order term for final states with several outgoing partons are tree-level matrix elements (no loops). If one expands the differential cross sections in different multiplicities one arrives at the following

$$\begin{aligned} d\sigma_0 &= C_0^{\mathrm{ME}}(\Omega_0)\, d\Omega_0 \\ d\sigma_1 &= C_1^{\mathrm{ME}}(\Omega_1)\, \alpha_{\mathrm{s}} d\Omega_1 \\ d\sigma_2 &= C_2^{\mathrm{ME}}(\Omega_2)\, \alpha_{\mathrm{s}}^2 d\Omega_2 \\ &\vdots \\ d\sigma_n &= C_n^{\mathrm{ME}}(\Omega_n)\, \alpha_{\mathrm{s}}^n d\Omega_n \\ &\vdots \end{aligned} \tag{2.7}$$

where we have used the short-hand notation

$$\Omega_n = (\mathbf{p}_1, \ldots, \mathbf{p}_{n+m}). \tag{2.8}$$

Here $\mathbf{p}$ is used to denote the momenta of the outgoing partons in the matrix element. The Born-level diagram considered has $m$ outgoing partons and $n$ denotes the number of extra outgoing partons. For simplicity we assume throughout that the Born-level cross sections does not contain any powers of $\alpha_{\mathrm{s}}$, but this requirement can easily be relaxed.

The tree-level expansion is divergent and one needs to introduce a phase-space cut to avoid collinear and soft configurations. Another important property of tree-level matrix elements is that they describe inclusive quantities. If one, *e.g.*, integrates the three-parton matrix element according to a jet definition, this yields the cross section for a configuration with *at least* three jets.

There a several methods for calculating tree-level matrix elements which have been implemented as parts of automated programs. These programs generate all the possible diagrams, sum them and generate events accordingly. Examples of such programs include MadEvent [24] and Alpgen [25].

There are many uncertainties associated with tree-level matrix element that can be better controlled if one could include the next order in the $\alpha_s$ expansion, which would mean including one-loop matrix elements. One of the problems with loop matrix elements is that they are infinite and frequently negative. Only the sum of the associated real emission with the virtual one is a finite quantity. In practice one assumes that emissions within a region are unresolved and that their amplitude can be added to the virtual contribution. There are a few choices to be made, depending on when to consider an emission unresolved and how to map the unresolved contribution onto the virtual one.

Most one-loop matrix elements are calculated with a method called Catani–Seymour dipole subtraction [26,27]. This method uses a function calculated analytically from dipoles that is added to the virtual contribution and subtracted from the real contribution. This way of calculating one-loop matrix elements has been proven to work quite well, but it needs to be modified to be applied in to our algorithm. The reason is that for our matching algorithms to work, a strict phase space cut is needed to separate resolved and unresolved emissions. This can be accomplished by modifying the subtraction scheme outside the singular regions.

If a jet cutoff, $y_{\mathrm{cut}}$, is used to determine when an emission is resolved, and the renormalization scale is set to $\mu$, one can formulate the cross sections for one-loop matrix elements in the following way.

$$
\begin{aligned}
d\sigma_0 &= C_0^{\mathrm{ME}}(\Omega_0)(1+\alpha_s(\mu)c_{0,1}^{\mathrm{ME}}(\Omega_0,y_{\mathrm{cut}}))\,d\Omega_0\\
d\sigma_1 &= C_1^{\mathrm{ME}}(\Omega_1)\,\alpha_s(\mu)(1+\alpha_s(\mu)c_{1,1}^{\mathrm{ME}}(\Omega_1,\mu,y_{\mathrm{cut}}))\Theta(y-y_{\mathrm{cut}})d\Omega_1\\
d\sigma_2 &= C_2^{\mathrm{ME}}(\Omega_2)\,\alpha_s^2(\mu)(1+\alpha_s(\mu)c_{2,1}^{\mathrm{ME}}(\Omega_2,\mu,y_{\mathrm{cut}}))\Theta(y-y_{\mathrm{cut}})d\Omega_2\\
&\;\vdots\\
d\sigma_{n-1} &= C_{n-1}^{\mathrm{ME}}(\Omega_{n-1})\,\alpha_s^{n-1}(\mu)(1+\alpha_s(\mu)c_{n-1,1}^{\mathrm{ME}}(\Omega_{n-1},\mu,y_{\mathrm{cut}}))\Theta(y-y_{\mathrm{cut}})d\Omega_{n-1}\\
d\sigma_n &= C_n^{\mathrm{ME}}(\Omega_n)\,\alpha_s^n(\mu)\Theta(y-y_{\mathrm{cut}})d\Omega_n
\end{aligned}
\tag{2.9}
$$

When one-loop matrix elements are calculated, the unresolved parton of the tree-level matrix element with one more outgoing particles needs to be added. The resolved part is considered to have a higher multiplicity but also needs to be included in the calculation. This means that the highest multiplicity is always calculated to tree-level accuracy.

The renormalization scale used in the calculation enters not only as a scale in $\alpha_s$, but also affects the one-loop terms (except for $c_{0,1}^{\mathrm{ME}}$ in case the Born level contains no powers of $\alpha_s$). It is quite simple to see how the renormalization scale enter if one considers the running coupling, which can be expanded as

$$
\begin{aligned}
\alpha_s(\mu') &= \alpha_s(\mu)\left(1+\alpha_s(\mu)\frac{\log(\mu/\mu')}{\alpha_0}+\mathcal{O}(\alpha_s^2(\mu))\right)\\
\alpha_0 &= \frac{2\pi}{\beta_0} = \frac{6\pi}{33-2n_f}
\end{aligned}
\tag{2.10}
$$

This means that a change in the renormalization scale of the first term leaves a remnant term in the next order in $\alpha_s$ simply through the running, which is something that needs to be taken into account if one varies the renormalization scale. The effect can be studied by expanding $\alpha_s$ explicity for one of the multiplicities

$$\alpha_s^l(\mu)(1+\alpha_s(\mu)c_{l,1}^{\mathrm{ME}}(\Omega_l,\mu,y_{\mathrm{cut}})) = \\ \alpha_s^l(\mu')(1+l\alpha_s(\mu')\frac{\log(\mu'/\mu)}{\alpha_0}+\alpha_s(\mu')c_{l,1}^{\mathrm{ME}}(\Omega_l,\mu,y_{\mathrm{cut}})+\mathcal{O}(\alpha_s^2(\mu'))), \tag{2.11}$$

which leads to the following scale dependence

$$c_{l,1}^{\mathrm{ME}}(\Omega_l,\mu',y_{\mathrm{cut}})) = c_{l,1}^{\mathrm{ME}}(\Omega_l,\mu,y_{\mathrm{cut}})+l\frac{\log(\mu'/\mu)}{\alpha_0}+\mathcal{O}(\alpha_s(\mu'))). \tag{2.12}$$

The renormalization scale can also be used to tune the different jet fractions, which is known as optimized perturbation theory [28]. Going beyond one-loop matrix elements or going to processes with incoming hadrons, there is also a renormalization scheme dependence to be considered.

### 2.3 Merging parton showers and tree-level matrix elements

The purpose of merging algorithms is to improve the description of jet observables without changing things such as the internal jet structure which is described well by the shower. At a more formal level the goal is to replace some coefficients in the expansion of the parton shower with their correct counterparts from the matrix elements.

There are several algorithms formulated with this purpose in mind. The main ones are CKKW [1, 2], CKKW-L [3, 4], MLM [5, 6] and Pseudo-Shower [7]. Their advantages and disadvantages when applied to $e^+e^-$ annihilation is discussed thoroughly in [29]. Here the discussion is limited to how the CKKW-L algorithm can be extended to also include one-loop matrix elements, but most of the general ideas can be applied also to CKKW.

Before going to one-loop matrix element, the mechanism used to merge tree-level matrix elements needs to be understood. With tree-level matrix elements two issues need to be resolved, namely to divide the phase space for emissions between the parton shower and the matrix element and to introduce Sudakov form factors to make the matrix elements exclusive. Essentially one would like to replace the product of splitting functions present in the cross sections for various processes in the parton shower with the correct tree-level matrix element.

The phase space is to be divided in such a way that the region for allowed emissions from the matrix element and the parton shower cover the entire phase space with no overlaps. Failing to do this consistently results in double counting or dead regions. The scale that defines the border between the matrix-element and the parton-shower phase space is known as the merging scale, and is usually defined using a jet clustering algorithm.

The Sudakov form factors are introduced by using a constructed shower history, which is done by considering all possible shower histories for the states generated according to the matrix element and selecting one with a probability proportional to the product of the corresponding splitting functions in the shower. The actual shower is then used for calculating the Sudakov form factors, which has the advantage that any non-leading effects that were introduced in the shower are also included. For further details on this procedure we refer the reader to [3, 4].

A particular shower scenario is dependent on the emission scales of the shower (denoted $\rho$) and other shower variables such as energy fraction and angular orientation, which are denoted simply by $\vec{x}$. A complete set of scales and other variables can be used to yield a shower history composed of specific states, denoted $S_i$. Looking at the differential exclusive cross section for $n$ emitted partons, the parton shower yields the following

$$\frac{d\sigma_n^{\mathrm{PS}}}{d\Omega_n^{\mathrm{PS}}} = KC_n^{\mathrm{PS}}(\Omega_n^{\mathrm{PS}})\Delta_{S_n}(\rho_n, \rho_c)\prod_{i=1}^{n}\alpha_{\mathrm{s}}(b\rho_i)\Delta_{S_{i-1}}(\rho_{i-1}, \rho_i), \tag{2.13}$$

where $\rho_0$ is the maximum scale, $\rho_c$ is the shower cutoff scale and $b$ is the parameter introduced in equation (2.5). An overall N$^l$LO $K$-factor, $K = 1 + \sum_{i=1}^{l} k_i\alpha_{\mathrm{s}}^i(\mu)$, has also been included.

The above expression is to be compared with the appropriate tree-level matrix element. The tree-level matrix element is an inclusive quantity and do not cover the full phase space, which is denoted by including a step function with the matrix element cutoff which is set equal to the merging scale, $y_{\mathrm{MS}}$. The cross section for the tree-level matrix element can be formulated in the following way.

$$\frac{d\sigma_n^{\mathrm{ME}}(y_{\mathrm{MS}})}{d\Omega_n} = C_n^{\mathrm{ME}}(\Omega_n)\alpha_{\mathrm{s}}^n(\mu)\Theta(y(\Omega_n) - y_{\mathrm{MS}}) \tag{2.14}$$

By selecting one history out of all possible histories, the matrix element can be mapped onto the shower phase space formulation, which can be described by the mapping $\Omega_n \mapsto \Omega_n^{\mathrm{PS}}$. The matrix element cross section can be written in terms of parton shower phase space in the following way.

$$\frac{d\sigma_n^{\mathrm{ME}}(y_{\mathrm{MS}})}{d\Omega_n^{\mathrm{PS}}} = C_n^{\mathrm{ME}}(\Omega_n^{\mathrm{PS}})\alpha_{\mathrm{s}}^n(\mu)\Theta(y(S_n) - y_{\mathrm{MS}}) \tag{2.15}$$

The Sudakov form factors are then introduced in the same way as in equation (2.13). In addition, the coupling constant is reweighted to use the emission scales instead of a fixed renormalization scale. The procedure results in the following exclusive cross section.

$$\begin{aligned}\frac{d\sigma_n(y_{\mathrm{MS}})}{d\Omega_n^{\mathrm{PS}}} = {} & KC_n^{\mathrm{ME}}(\Omega_n^{\mathrm{PS}})\alpha_{\mathrm{s}}^n(\mu)\Theta(y(S_n) - y_{\mathrm{MS}})\Delta_{S_n}(\rho_n, \rho_c) \times \\ & \prod_{i=1}^{n}\frac{\alpha_{\mathrm{s}}(b\rho_i)}{\alpha_{\mathrm{s}}(\mu)}\Delta_{S_{i-1}}(\rho_{i-1}, \rho_i)\end{aligned} \tag{2.16}$$

This expression is fully exclusive in the same way as a state generated by the shower. However, the tree-level matrix element is only allowed to generate emissions above the merging scale. The scheme therefore needs to be supplemented by introducing a way of allowing the shower to generate extra emissions below the merging scale and the methods for accomplishing this is the subject of the next section.

### 2.4 Adding parton showers to multi-parton states

If the merging scale, $y_{\rm MS}$, is defined in the same way as the parton shower ordering variable, the adding of a parton shower is fairly trivial. For most parton showers you can simply shower each parton individually and use $y_{\rm MS}$ as the maximum scale for the ordering variable.

If, however, the merging scale and ordering scale are different, *e.g.* the merging scale is defined in invariant mass, while the shower ordering is in transverse momentum (as is the case in our studies below in section 4), the problem becomes non-trivial.

In the original CKKW formulation, the problem was solved by introducing the concept of a "vetoed shower". Here, each parton is allowed to shower, starting from a value of the ordering variable typically given by the maximum possible scale in the Born-level process. Each emission is then checked so that it if is above the merging scale, $y_{\rm MS}$, the emission is discarded, allowing the shower to continue to evolve down to lower evolution scales.

There is, however, a problem with this procedure, as was noted in [13] and [29], related to the fact that the shower may be allowed to make effectively unordered emissions. To understand the problem, we consider a partonic state corresponding to $n$ parton emissions beyond the Born level. The state can be mapped onto a set of intermediate states and scales, where $\rho_n$ represents the scale of the last emission. Now if the merging scale is very different from the ordering scale, it may very well happen in a vetoed shower that an emission with $y < y_{\rm MS}$ and $\rho > \rho_n$ is generated, which breaks the ordering, since in the shower such an emission should have been emitted from an intermediate state. Breaking the ordering results in the wrong colour structure, which may result in incorrect treatment of coherence effect, and different kinematics, which may give unwanted suppressions in some regions of phase space.

So far, two solutions to this problem has been presented. The CKKW-L approach and the so-called "truncated" vetoed shower. Both approaches requires that not only emission scales are reconstructed as in CKKW, but also the full kinematics of the complete shower history with on-shell intermediate states, $S_i$.

The truncated shower [13][2] is way of allowing the shower to generate emissions from the intermediate states in the shower history and thereby preserve the ordering. The vetoed shower is started from the Born-level state, $S_0$ with the corresponding maximum scale $\rho_0$ and is vetoed in the same way as above. When the shower evolution comes down to $\rho_1$, it is stopped and the reconstructed emission $(\rho_1, \vec{x}_1)$ that was generated by the matrix element is inserted by hand. The vetoed evolution is then continued down to $\rho_2$, where the next reconstructed emission is inserted, and so on, in a way such that the kinematics of the partons in the original state is minimally disturbed.

The philosophy of the CKKW-L approach is quite different, in that unordered emissions are simply forbidden. This means that as soon as there is one emission below the merging scale, that emission and all subsequent emission (above and below the merging scale) are generated by the shower. When adding a shower to a $n$-parton state with the reconstructed scale $\rho_n$, the shower is started from $\rho_n$ and the first emission is forced to be below the

[2]Here we only give our rough interpretation of the truncated shower. For a more detailed description we refer to [13]

merging scale, but later emissions have no such restrictions. The reweighting for the same state is done by using the full Sudakov form factor down to the scale $\rho_n$. The state which are thus forbidden can be generated by matrix elements for lower multiplicity together with the shower.

The CKKW-L approach can be applied to the cross section in equation (2.16) modifying the last Sudakov form factor and adding a shower. The first emission is added with the restriction that it should be below $\rho_n$ and the merging scale. The emission probability can be described as

$$dP = \alpha_{\rm s}(b\rho)\Gamma_{S_n}(\rho,\vec{x})\Theta(y_{\rm MS} - y(S_n,\rho,\vec{x}))\Delta_{S_n}(\rho_n,\rho;< y_{\rm MS})d\rho d\vec{x}, \tag{2.17}$$

where

$$\Delta_{S_n}(\rho_n,\rho;< y_{\rm MS}) = \exp\left(-\int_{\rho}^{\rho_n} d\rho' d\vec{x}\alpha_{\rm s}(b\rho')\Gamma_{S_i}(\rho',\vec{x})\Theta(y_{\rm MS} - y(S_n,\rho',\vec{x}))\right). \tag{2.18}$$

In addition to the probability described above the CKKW-L procedure specifies that one should also include the Sudakov form factor between the generated emission $\rho_{n+1}$ and the previous emission $\rho_n$ and above the merging scale, $\Delta_{S_n}(\rho_n,\rho_{n+1};> y_{\rm MS})$. The cascade is then continued from $\rho_{n+1}$ with no veto.

An alternative way of performing the same calculation is to generate one emission starting from $\rho_n$ and discard the entire event if the emission was above the merging scale. It is equivalent to the above procedure since the probability for discarding the event is equal to the Sudakov form factor calculated at the end. This is the way the algorithm was formulated in [3]

The exclusive cross section in equation (2.16) should be modified accordingly. Assuming that one emission has been added using the procedure describe above, one can formulate the cross section in the following way.

$$\frac{d\sigma_n(y_{\rm MS})}{d\Omega_n^{\rm PS}} = KC_n^{\rm ME}(\Omega_n^{\rm PS})\alpha_{\rm s}^n(\mu)\Theta(y(S_n) - y_{\rm MS})\Delta_{S_n}(\rho_n,\rho_{n+1};> y_{\rm MS}) \times \prod_{i=1}^{n}\frac{\alpha_{\rm s}(b\rho_i)}{\alpha_{\rm s}(\mu)}\Delta_{S_{i-1}}(\rho_{i-1},\rho_i) \tag{2.19}$$

Another important issue to consider when adding a parton shower is how to handle the highest multiplicity states with $n = N$. Clearly we must here not veto emissions above $y_{\rm MS}$, since this would artificially suppress final states with more than $N$ emissions above this scale. In this case no extra Sudakov form factors needs to be included and the cross section is given by

$$\frac{d\sigma_n(y_{\rm MS})}{d\Omega_n^{\rm PS}} = KC_n^{\rm ME}(\Omega_n^{\rm PS})\alpha_{\rm s}^n(\mu)\Theta(y(S_n) - y_{\rm MS})\prod_{i=1}^{n}\frac{\alpha_{\rm s}(b\rho_i)}{\alpha_{\rm s}(\mu)}\Delta_{S_{i-1}}(\rho_{i-1},\rho_i). \tag{2.20}$$

The only constraint on the shower is that the first emission should be below $\rho_n$.

In the CKKW-L algorithm, emissions are thus corrected with the full matrix element only if they are among the $N$ hardest (according to the parton shower ordering) *and* are all above the merging scale.

### 2.5 Extending to one-loop MEs

To be able to extend the algorithm to also include one-loop matrix elements a new set of issues has to be addressed. First and foremost, the one-loop matrix element contains a terms which is one order higher in $\alpha_{\rm s}$. To be able to apply a correction, the shower cross section, eq. (2.13), therefore needs to be expanded to that level, with a fixed renormalization scale $\mu$. The $\alpha_{\rm s}$ expansion of the Sudakov form factor and the running coupling can be written

$$\begin{aligned}\Delta_{S_i}(\rho_i,\rho_{i+1}) &= 1-\int_{\rho_{i+1}}^{\rho_i} d\rho\alpha_{\rm s}(b\rho)\Gamma^{\rm PS}_{S_i}(\rho)+\ldots \\ &= 1-\alpha_{\rm s}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho\Gamma^{\rm PS}_{S_i}(\rho)+\mathcal{O}(\alpha_{\rm s}^2(\mu)) \qquad (2.21)\\ \alpha_{\rm s}(b\rho) &= \alpha_{\rm s}(\mu)\left(1+\alpha_{\rm s}(\mu)\frac{\log(\mu/(b\rho))}{\alpha_0}+\mathcal{O}(\alpha_{\rm s}^2(\mu))\right), \qquad (2.22)\end{aligned}$$

where the possibility of modifying the scale used in $\alpha_{\rm s}$ in the shower has been included from equation (2.5).

This means the parton shower cross section in eq. (2.13) can be rewritten as

$$\begin{aligned}\frac{d\sigma_n^{\rm PS}}{d\Omega_n^{\rm PS}} = C_n^{\rm PS}(\Omega_n^{\rm PS})\alpha_{\rm s}^n(\mu)\Bigg[1+\alpha_{\rm s}(\mu)\Bigg\{k_1+\sum_{i=1}^{n}\frac{\log(\mu/(b\rho_i))}{\alpha_0} \\ -\sum_{i=0}^{n-1}\int_{\rho_{i+1}}^{\rho_i} d\rho\Gamma_{S_i}(\rho)-\int_{\rho_c}^{\rho_n} d\rho\Gamma_{S_n}(\rho)\Bigg\}+\mathcal{O}(\alpha_{\rm s}^2(\mu))\Bigg]. \qquad (2.23)\end{aligned}$$

Note that the extra term that appears because of the running coupling is equivalent to the term which appears if the renormalization scale of the one-loop matrix element is changed, derived explicity in equation (2.12).

The one-loop matrix elements need to be put through the procedure described earlier to construct a shower history. The procedure yields the following form for the cross section.

$$\frac{d\sigma_n^{\rm ME}(y_{\rm MS})}{d\Omega_n^{\rm PS}} = C_n^{\rm ME}(\Omega_n^{\rm PS})\alpha_{\rm s}^n(\mu)\left[1+\alpha_{\rm s}(\mu)c_{n,1}^{\rm ME}(\Omega_n^{\rm PS},\mu,y_{\rm MS})\right]\Theta(y(S_n)-y_{\rm MS}) \qquad (2.24)$$

The cross section contains the same terms as the tree-level cross section plus the next order correction to the matrix element, which is a sum of the virtual diagrams of multiplicity $n$ and the real diagrams with multiplicity $n+1$ kinematics below the merging scale $y_{\rm MS}$.

If the merging scale is defined using the same scale as the ordering variable in the shower, no further modifications to the matrix element would have been necessary. However, to preserve the ordering of the shower, extra Sudakov form factors need to be included. The phase space in question is where emissions have a higher ordering variable ($\rho$) than the last emission of the matrix element ($\rho_n$) and are below the merging scale. These emissions would violate the shower ordering and are therefore forbidden, but one still needs to include virtual corrections in this region, which is done using Sudakov form factors. The reweighted one-loop cross sections can be written as

$$\frac{d\sigma_n(y_{\rm MS})}{d\Omega_n^{\rm PS}} = C_n^{\rm ME}(\Omega_n^{\rm PS})\alpha_{\rm s}^n(\mu)\left[1+\alpha_{\rm s}(\mu)c_{n,1}^{\rm ME}(\Omega_n^{\rm PS},\mu,y_{\rm MS})\right]\Theta(y(S_n)-y_{\rm MS})\times$$
$$\prod_{i=0}^{n-1}\Delta_{S_i}(\rho_i,\rho_{i+1};<y_{\rm MS}), \tag{2.25}$$

where the definition for the Sudakov from factor was presented in equation (2.18). Note that no reweighting is necessary for the lowest multiplicity processes, since if there are no emissions in the matrix element state, there are no regions of phase space where the shower can generate emissions violating the ordering requirement.

The other component that goes into the merging is the shower where all the terms corresponding to the one-loop matrix element have been subtracted. When doing the subtraction we choose to work with a cascade which is already corrected with tree-level matrix elements. The idea is to reweight the tree-level matrix element the same way as CKKW-L, which is described by equation (2.19), and then subtract the terms corresponding to the one-loop matrix element. Note that one of the Sudakov form factors has a dependency on the scale of the next emission performed by the shower $\rho_{n+1}$, which was described in section 2.4.

The terms of order $\alpha_{\rm s}^{n+1}$ in equation (2.23) needs to modified to comply with the phase space restrictions of the one-loop matrix element. The shower is formulated in a way where emissions can be anywhere within the allowed shower phase space, whereas the matrix element is restricted to emissions above the merging scale. This difference in phase space does not affect the expansion of the running coupling, but must be included in the integration over the branching probability. The full formula including tree-level matrix element corrected with CKKW-L and the subtraction of the one-loop terms is the following.

$$\begin{aligned}\frac{d\sigma_n^{\rm PScorr}(y_{\rm MS})}{d\Omega_n^{\rm PS}} = {} & C_n^{\rm ME}(\Omega_n^{\rm PS})\alpha_{\rm s}^n(\mu)\times\\
& \Bigg[K\Delta_{S_n}(\rho_n,\rho_{n+1};>y_{\rm MS})\prod_{i=1}^{n}\frac{\alpha_{\rm s}(b\rho_i)}{\alpha_{\rm s}(\mu)}\Delta_{S_{i-1}}(\rho_{i-1},\rho_i)\\
& \quad -\prod_{i=0}^{n-1}\Delta_{S_i}(\rho_i,\rho_{i+1};<y_{\rm MS})\Bigg\{1+k_1\alpha_{\rm s}(\mu)+\alpha_{\rm s}(\mu)\sum_{i=1}^{n}\frac{\log(\mu/(b\rho_i))}{\alpha_0}\\
& \quad -\alpha_{\rm s}(\mu)\sum_{i=0}^{n-1}\int_{\rho_{i+1}}^{\rho_i}d\rho d\vec{x}\,\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x})-y_{\rm MS})\\
& \quad -\alpha_{\rm s}(\mu)\int_{\rho_c}^{\rho_n}d\rho d\vec{x}\,\Gamma_{S_n}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x})-y_{\rm MS})\Bigg\}\Bigg]\end{aligned} \tag{2.26}$$

The samples described by eq. (2.25) and eq. (2.26) are added together in the end to form the parton multiplicity cross section for one-loop matrix elements merged together with parton showers. When implementing the algorithms one also allows for extra emissions generated by the shower with the requirement that the first such emission should have a lower scale than $\rho_n$ and be below the merging scale. The details of the entire procedure is described in section 3.

### 2.6 Two partons at two loops

In the case of $e^+e^- \to$ hadrons, there is an additional matrix element that can be included without considering higher order terms in the running of $\alpha_s$, which is the two-loop matrix element for two partons. The reason is that the leading-order term for the two-parton cross section does not include $\alpha_s$. The matrix element is also fairly easy to simulate if one knows the other components at order $\alpha_s^2$. Working with two-loop terms means that the equation (2.21) needs to be expanded one more order including the parameter $b$, defined in equation (2.5). The expanded Sudakov form factor can be written in the following way.

$$\begin{aligned}\Delta_{S_i}(\rho_i,\rho_{i+1}) &= 1-\int_{\rho_{i+1}}^{\rho_i} d\rho\alpha_s(b\rho)\Gamma_{S_i}^{\mathrm{PS}}(\rho)+\frac{1}{2!}\left(\int_{\rho_{i+1}}^{\rho_i} d\rho\alpha_s(b\rho)\Gamma_{S_i}^{\mathrm{PS}}(\rho)\right)^2-\dots\\ &= 1-\alpha_s(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho\Gamma_{S_i}^{\mathrm{PS}}(\rho)-\alpha_s^2(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho\frac{\log(\mu/(b\rho))}{\alpha_0}\Gamma_{S_i}^{\mathrm{PS}}(\rho)+\\ &\quad \frac{1}{2}\alpha_s^2(\mu)\left(\int_{\rho_{i+1}}^{\rho_i} d\rho\Gamma_{S_i}^{\mathrm{PS}}(\rho)\right)^2+\mathcal{O}(\alpha_s^3(\mu))\end{aligned}\tag{2.27}$$

Apart from the terms above, the $K$-factor of order $\alpha_s^2$ needs to be included, but for the running $\alpha_s$ it is sufficient to include the first order expansion of $\alpha_s$ given by equation (2.22). The tree-level matrix element can be modified in the following fashion, which is to be added to the event generated according to the two loop matrix element.

$$\begin{aligned}\frac{d\sigma_2^{\mathrm{PScorr}}(y_{\mathrm{MS}})}{d\Omega_0^{\mathrm{PS}}} = C_0^{\mathrm{ME}}(\Omega_0^{\mathrm{PS}})\Bigg[K\Delta_{S_0}(\rho_0,\rho_1;>y_{\mathrm{MS}})-\Bigg\{&1+k_1\alpha_s(\mu)+k_2\alpha_s^2(\mu)\\ &-\alpha_s(\mu)(1+k_1\alpha_s(\mu))\int_{\rho_c}^{\rho_0} d\rho d\vec{x}\,\Gamma_{S_0}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x})-y_{\mathrm{MS}})\\ &-\alpha_s^2(\mu)\int_{\rho_c}^{\rho_0} d\rho d\vec{x}\,\frac{\log(\mu/\rho)}{\alpha_0}\Gamma_{S_0}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x})-y_{\mathrm{MS}})\\ &+\frac{\alpha_s^2(\mu)}{2}\left(\int_{\rho_c}^{\rho_0} d\rho d\vec{x}\,\Gamma_{S_0}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x})-y_{\mathrm{MS}})\right)^2\Bigg\}\Bigg]\end{aligned}\tag{2.28}$$

This term represents the modified parton shower and added to the two-parton matrix element calculated at two loops, the cascade becomes corrected at one order higher in $\alpha_s$. Note that there is a dependence on the scale of the next emission generated by the shower, in this case $\rho_1$, in the same way as in equations (2.19) and (2.26).

The term described above should be added to the two-parton matrix element calculated at two-loop accuracy. Since the two-loop matrix element considered has the lowest multiplicity, no reweighting due to the ordering requirement is necessary. If two-loop matrix elements with higher parton multiplicity were to be included, they would have to be reweighted in the same way as the one-loop matrix elements, described in eq. (2.25).

## 3. The algorithm

This section describes all the necessary steps needed to generate the actual events. First the methods for calculating each individual weight is presented and then each step in the algorithm is described.

### 3.1 Calculating the terms

Each of the weights that need to be calculated consists of a number of common elements. This section describes how to calculate each individual piece. All the terms are calculated similarly to the CKKW-L approach, which means that the actual shower is used. However, it should be noted that it is possible to do calculations along the same lines using the analytical weights in the CKKW algorithm. Using the actual shower puts a constraint on which parton showers can be used. The algorithm can only be applied to showers with well defined intermediate states and Sudakov form factors that factorizes, which is the same requirement as the CKKW-L algorithm.

The weights described in this section are calculated using Monte Carlo techniques, which means that there is an element of randomness in each weight. The calculations presented give the correct value for each weight if an average value is calculated. In the general framework of Monte Carlo event generators this does not present a problem.

The simplest weight to be calculated is the plain Sudakov form factor, denoted $\Delta_{S_i}(\rho_i, \rho_{i+1})$. This is done by generating one emission with the shower starting from the state $S_i$ and the scale $\rho_i$. If the emission is above the scale $\rho_{i+1}$ set the weight to zero otherwise set it equal to one. This gives the correct behavior since by definition the no-emission probability of the shower is equal to the Sudakov form factor.

The next weight to be described is the Sudakov form factor that is modified to exclude emissions above the merging scale denoted $\Delta_{S_i}(\rho_i, \rho_{i+1}; < y_{\text{MS}})$ and defined in equation (2.18), and is used to reweight the states generated according to one-loop matrix elements. In this case two different scales are mixed and the scheme for the plain Sudakov form factor needs to be supplemented with an accept/reject scheme. The calculation is described in the following steps.

1. Feed the state $S_i$ into the shower and generate one emissions starting from a scale $\rho_i$. This yields an emission scale $\rho$ and a new state $S$.
2. Depending on the emission there are three options.
   - If the scale of the generated emission $\rho$ is above $\rho_{i+1}$ and if the emission is above the merging scale ($y(S) > y_{\text{MS}}$) generate a new emission from the state $S_i$, but this time using $\rho$ as the starting scale. Repeat step 2.
   - If the scale of the generated emission $\rho$ is below $\rho_{i+1}$ set the weight to one.
   - If the scale of the generated emission $\rho$ is above $\rho_{i+1}$ and if the emission is below the merging scale ($y(S) < y_{\text{MS}}$) set the weight to zero.

The only other terms that need to be calculated are the integrated branching probabilities. First the calculation of the term $\alpha_{\text{s}}(\mu) \int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\, \Gamma_{S_i}(\rho, \vec{x}) \Theta(y(S_i, \rho, \vec{x}) - y_{\text{MS}})$ is described and later it is shown how to extend the calculation to the other related terms.

The calculation is done by generating emissions in the entire avaliable phase space and counting the total number. The procedure is described in the following steps.

1. Use a fixed value for $\alpha_\mathrm{s} = \alpha_\mathrm{s}(\mu)$ (usually avaliable as an option in a parton shower program) and generate one emission from the state $S_i$ with a starting scale $\rho_i$. This yields an emission scale $\rho$ and a new state $S$.
2. If the emission has a scale $\rho > \rho_{i+1}$ and is above the merging scale $y(S) > y_\mathrm{MS}$ count the emission.
3. Depending on the scale of the emission stop the algorithm or generate another emission.
   - If the scale of the generated emission $\rho$ is above $\rho_{i+1}$ generate a new emission from the state $S_i$, but this time using $\rho$ as the starting scale. Repeat step 2 and 3.
   - If the scale of the generated emission $\rho$ is below $\rho_{i+1}$ set the weight to the total number of emission counted in step 2.

The average number of emissions in the algorithm above gives the correct value for the integral. To show that this is the case consider the probability of $n$ emissions.

$$P(n) = \Delta_{S_i}(\rho_i, \rho_{i+1}; > y_\mathrm{MS})\frac{1}{n!}\left(\alpha_\mathrm{s}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x}) - y_\mathrm{MS})\right)^n. \quad (3.1)$$

The average number of emission can be written as

$$\begin{aligned}\sum_{n=0}^{\infty} nP(n) &= \Delta_{S_i}(\rho_i, \rho_{i+1}; > y_\mathrm{MS})\alpha_\mathrm{s}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x}) - y_\mathrm{MS}) \times \\ &\sum_{n=1}^{\infty}\frac{1}{(n-1)!}\left(\alpha_\mathrm{s}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x}) - y_\mathrm{MS})\right)^{n-1} \\ &= \alpha_\mathrm{s}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x}) - y_\mathrm{MS}). \end{aligned} \quad (3.2)$$

The same algorithm can also be used to calculate the higher order terms present in equation (2.28). The integrated branching probability squared can be calculated by taking setting the weight to $n(n-1)$, where $n$ is the number of emissions, which gives the correct average since

$$\sum_{n=0}^{\infty} n(n-1)P(n) = \left(\alpha_\mathrm{s}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x}) - y_\mathrm{MS})\right)^2. \quad (3.3)$$

There is one more term to be considered in equation (2.28), which is

$$\alpha_\mathrm{s}^2(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\frac{\log(\mu/(b\rho))}{\alpha_0}\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x}) - y_\mathrm{MS}). \quad (3.4)$$

The way this integral is calculated is by an accept/reject scheme. The first step is to rewrite the term where the value of the logarithm has been divided by its maximum, which occurs for the minimum possible value of $\rho$.

$$\left(\frac{\alpha_{\mathrm{s}}(\mu)\log(\mu/(b\rho_{i+1}))}{\alpha_0}\right)\left(\alpha_{\mathrm{s}}(\mu)\int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\,\frac{\log(\mu/(b\rho))}{\log(\mu/(b\rho_{i+1}))}\Gamma_{S_i}(\rho,\vec{x})\Theta(y(S_i,\rho,\vec{x})-y_{\mathrm{MS}})\right) \tag{3.5}$$

The factor to the right can now be simulated with a scheme simular to the one described earlier. The only addition is that emissions are only counted if $\log(\mu/(b\rho))/\log(\mu/(b\rho_{i+1})) > R$, where $R$ is a random number between 0 and 1.

### 3.2 The steps

To generate the actual events we start with two different samples. One is generated according to the one-loop matrix element and one generated with tree-level matrix elements, where both samples are generated using the same cutoff, $y_{\mathrm{MS}}$. The two samples are generated for all multiplicates except for the highest one, where only the tree-level matrix element is used. Different weights are calculated for the events depending on if they were generated according to a one-loop matrix element or a tree-level matrix element. Event are generated according to the following steps:

1. Choose a merging scale $y_{\mathrm{MS}}$ and use the same scale as matrix element cutoff. Calculate the cross section for the one-loop matrix element with multiplicities $n < N$ and for tree-level matrix elements with multiplicities $n \leq N$. Choose a matrix element with a probability proportional to its cross section.
2. Generate an event with a kinematic distribution in accordance with the chosen matrix element.
3. Construct a shower history by considering all possible histories and selection one with a probability proportional to the corresponding product of splitting functions. This leads to a set of states $S_n \ldots S_0$ and scales $\rho_n \ldots \rho_1$.
4. When generating the first emission from the shower, there are two cases to be considered.
   - If the event was generated according to a one-loop matrix element or according to a tree-level matrix element with a multiplicity less than the maximum ($n < N$), generate one emission starting from the state $S_n$ with a starting scale $\rho_n$, but veto any emission which is above the merging scale $y_{\mathrm{MS}}$
   - If the event was generated according to a tree-level matrix element and had the highest multiplicity ($n = N$), generate one emission from the state $S_n$ with a starting scale $\rho_n$.
5. The events are reweighted depending on type:
   - If the event was generated according to a one-loop matrix element, reweight the event with a factor $\prod_{i=0}^{n-1}\Delta_{S_i}(\rho_i,\rho_{i+1};<y_{\mathrm{MS}})$ according to the steps in subsection 3.1.
   - If the event was generated according to a tree-level matrix element, but did not have the highest multiplicity ($n < N$), then the the weight depends on the

scale of the emission in step 4, $\rho_{n+1}$. (If the shower cutoff was reached and no emission generated, set $\rho_{n+1} = \rho_c$.) Reweight the event with

$$
\begin{aligned}
&K\Delta_{S_n}(\rho_n, \rho_{n+1}; > y_{\mathrm{MS}}) \prod_{i=1}^{n} \frac{\alpha_s(b\rho_i)}{\alpha_s(\mu)} \Delta_{S_{i-1}}(\rho_{i-1}, \rho_i) \\
&- \prod_{i=0}^{n-1} \Delta_{S_i}(\rho_i, \rho_{i+1}; < y_{\mathrm{MS}}) \left\{ 1 + k_1 \alpha_s(\mu) + \alpha_s(\mu) \sum_{i=1}^{n} \frac{\log(\mu/(b\rho_i))}{\alpha_0} \right. \\
&\qquad - \alpha_s(\mu) \sum_{i=0}^{n-1} \int_{\rho_{i+1}}^{\rho_i} d\rho d\vec{x}\, \Gamma_{S_i}(\rho, \vec{x}) \Theta(y(S_i, \rho, \vec{x}) - y_{\mathrm{MS}}) \\
&\qquad \left. - \alpha_s(\mu) \int_{\rho_c}^{\rho_n} d\rho d\vec{x}\, \Gamma_{S_i}(\rho, \vec{x}) \Theta(y(S_i, \rho, \vec{x}) - y_{\mathrm{MS}}) \right\}, \qquad (3.6)
\end{aligned}
$$

according to the steps in subsection 3.1.

- If the event was generated according to a tree-level matrix element, but had the highest multiplicity ($n = N$), then the event is reweighted by $\prod_{i=1}^{n} \frac{\alpha_s(b\rho_i)}{\alpha_s(\mu)} \Delta_{S_{i-1}}(\rho_{i-1}, \rho_i)$.

6. Continue the cascade below $\rho_{n+1}$

## 4. Results

Our algorithm has been implemented using ARIADNE version 4.12 [30], which has been modified to include the possibility of calculating the different weights needed. The matrix elements used is taken from an implementation in PYTHIA version 6.414 [3] [31], where the $e^+e^-$ matrix elements were calculated in [32] and parameterized in [33]. The implementation includes the possibility of generating $e^+e^-$ events according to zero, one, or two orders in $\alpha_s$. This means that the four jets can only be generated according to the tree-level matrix element, whereas the three jet contribution can be generated using the one-loop contribution and the two jet can include up to two loops. The different multiplicities are separated using a cutoff in invariant mass divided by center of mass energy ($Q^2/s$), which can be varied between 0.01 and 0.05. The process considered throughout this section is $e^+e^-$ to hadrons at the $Z^0$ mass peak.

The first thing to be studied is how the algorithm behaves for the somewhat trivial case of calculating three partons with a tree-level matrix element and two partons to one loop. To study the effects of the cutoff, the JADE [34] jet clustering algorithm is used, since it has a jet scale which closely resembles the scale used for the cutoff, $y_{\mathrm{MS}}$. The matrix elements are calculated using a fixed value of the $\alpha_s$ used in ARIADNE at the renormalization scale, which is set to $m_Z$ (*i.e.* not using eq. (2.5) and setting $b = 1$ in the rest of section 2 and 3). The distribution in clustering scale for the third jet for our new procedure[4] is shown in figure 1 with three different values of the merging scale, 0.01, 0.02 and 0.05 and is compared

[3] PYTHIA has been modified to allow renormalization scales bigger than the center of mass energy and to return negative weights instead of rounding to zero for three partons at one loop.

[4] The results for our new procedure is throughout denoted NL$^3$.

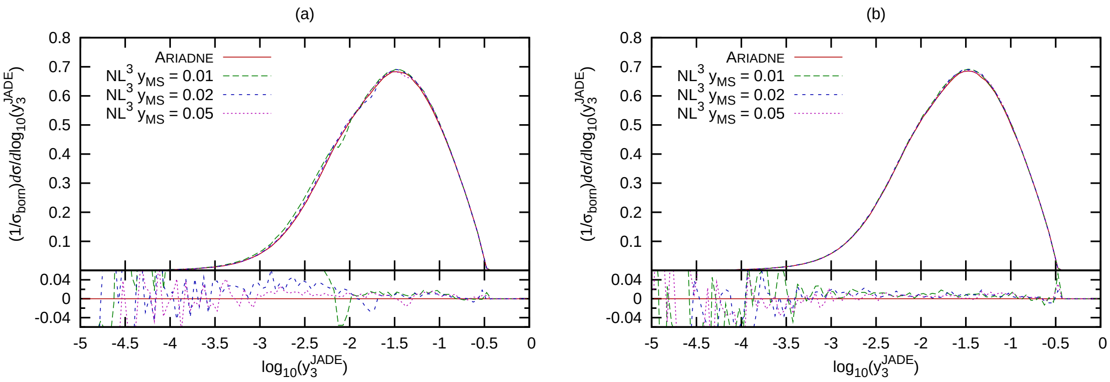


**Figure 1:** The parton-level $y_3$ spectra according to the JADE definition for the samples corrected with one-loop two-parton matrix element and tree-level three-parton matrix element. (a) includes all flavours and masses and (b) is with only massless d-quarks. The in-sets at the bottom of the plots show the relative differences between the results from NL$^3$ and default ARIADNE, $(\sigma_{\mathrm{NL}^3} - \sigma_{\mathrm{ARIADNE}})/\sigma_{\mathrm{ARIADNE}}$.

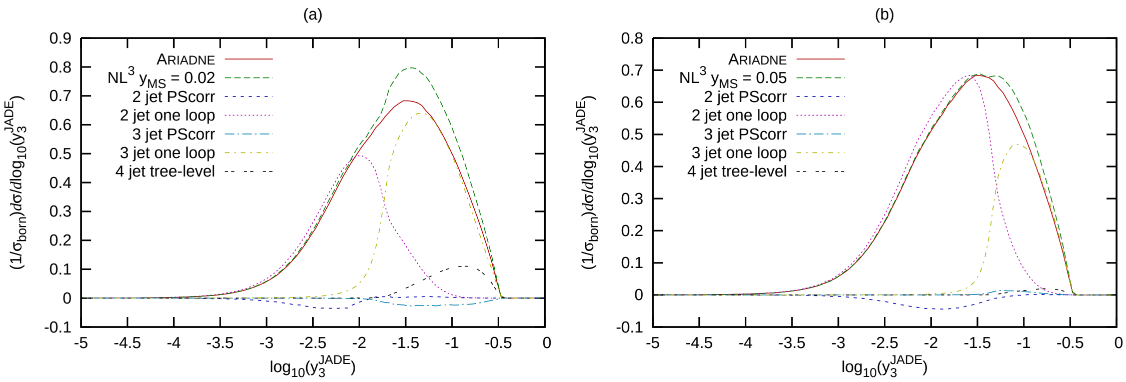


**Figure 2:** The parton-level $y_3$ spectra according to the JADE definition for the samples corrected with matrix elements describing two partons at one loop, three partons at one loop and four partons at tree level. (a) shows the curves for a cutoff of 0.02 and (b) for a cutoff 0.05. Both figures include curves displaying the various components of the NL$^3$ samples.

to the standard ARIADNE shower. The figure includes one plot with all flavours and masses and one where only massless d-quarks was used. Including quark masses there are some deviations, especially close to the cutoff. This is due to slightly different treatments of the suppression of radiation from heavy quarks (the dead-cone effect). The matrix element in PYTHIA uses the exact formula, whereas in ARIADNE a more general approximate formula is used. However, when masses are not included no deviations are visible, which should be the case since both the $K$-factor and the matrix element correction are present in the shower to this order.

Moving on to the simplest non-trivial case which includes two partons to one loop, three partons to one loop and four partons to tree level. Also here the clustering scale of the third jet using JADE is studied for the same $\alpha_{\mathrm{s}}$, renormalization scale and merging scales. The distribution is shown in figure 2 together with the various components that make up each distribution. The cutoff 0.01 is not included since it would have a negative two-parton cross section, which is not allowed in the PYTHIA routines.

The different components in figure 2 can be identified in the following way. The curves marked *one loop* are simply the contributions from the one-loop matrix elements with a Sudakov form factor according to equation (2.25) and a shower added below the merging scale, the curves marked *PScorr* are the contribution calculated from the tree-level matrix elements according to equation (2.26) and the curve marked *4 jet tree-level* is the highest multiplicity contribution which is calculated according to equation (2.20). The dominant contributions are clearly the one-loop matrix elements for two-parton and three-parton configurations. The four-parton matrix element is also significant at the hard end of the spectrum, but the contributions from the modified tree-level distributions are generally small.

We also note that the modified tree-level contribution (*PScorr*) have a slightly negative value. This is a result of the expansion of the Sudakov form factor together with the running coupling. However, negative weights can be avoided if one chooses a merging scale defined using the ordering variable in the shower, and a renormalization scale equal to the merging scale. As long as the one-loop matrix elements are not themselves negative, which happens for small enough merging scales, all events would then have positive weights.

One important feature of the algorithm is the possibility of including several multiplicities together. The importance of this is illustrated in figure 2 by the fact that the two-parton components give a contribution which extend significantly above the merging scale (while the opposite is true for the three parton contribution). Clearly it would be problematic to try to describe the jet distribution using only three- and four-parton matrix elements, although the calculation would be formally correct to NLO accuracy.

Another thing that is noticeable is that there is a significant overshoot above the cutoff in figure 2. This is attributed to the one-loop term of the three parton matrix element, which can not be accurately reproduced in the cascade. The equivalent term in the cascade is calculated using the Sudakov form factors and the value is significantly smaller than the matrix element.

The whole issue of the one-loop contribution being significantly larger than the parton shower counterpart has another consequence, namely that the $\alpha_{\rm s}$ used in the shower is higher than what is fitted to precision calculations, which include both fixed-order matrix elements and logarithmic resummations. However, simply lowering $\alpha_{\rm s}$ everywhere would destroy the agreement between the curves below the cutoff. We therefore both lower the value of $\alpha_{\rm s}$ and modify the scale used as a argument in the shower by using the parameter $b$ (defined in equation (2.5)). The value of $\alpha_{\rm s}$ from the PDG [35] which is $\alpha_{\rm s}(m_Z) = 0.1176$ which corresponds to $\Lambda_{\rm QCD} = 85.8$ MeV (assuming a leading order $\alpha_{\rm s}$ which is used in ARIADNE), which should be compared to the ARIADNE default $\Lambda_{\rm QCD} = 220$ MeV. Figure 3 shows the curves using the PDG value for $\alpha_{\rm s}$ and $b = 85.8/220 = 0.389$. There is a much better agreement for values above the cutoff, but there a still some discrepancies.

For the two-parton matrix element the calculation is also available at two loops. This was discussed in general in section 2.6. In figure 4 the two-loop corrections have been

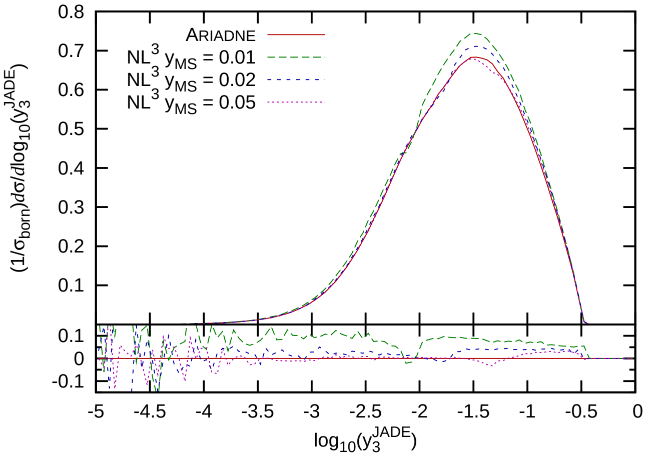


**Figure 3:** The parton-level $y_3$ spectra according to the JADE definition for the NL$^3$ algorithm including a modified way of treating $\alpha_s$.

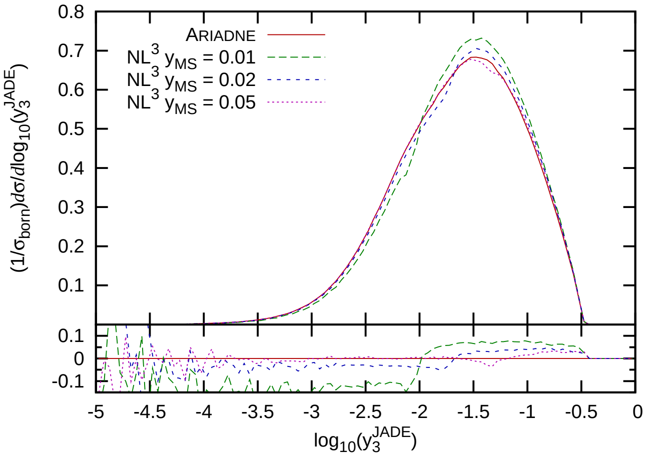


**Figure 4:** The parton-level $y_3$ spectra according to the JADE definition for the NL$^3$ algorithm including a modified way of treating $\alpha_s$ and the two-parton matrix element at two loops.

included, still using the same $\alpha_s$ treatment as described above. There is a clear difference in that the curves no longer overlap with ARIADNE for values below the merging scale. This happens since the two-loop contribution is beyond what can be reproduced by ARIADNE.

To check the consistency of the algorithm we have studied the sensitivity to changes in the renormalization scale. In figure 5 the renormalization scale has been varied up and down by a factor of two for both the case of one-loop correction and including two partons at two loops. Overall, the sensitivity to changes in the renormalization scale is small, with variations of around two percent in the results. This is expected since all the higher order terms comes from the shower which is unaffected by the renormalization scale.

Finally our algorithm is compared to data from the DELPHI [36] experiment. It should be noted that all the data are quite well reproduced by ARIADNE and since we get small deviations in the previous plot, we expect small differences here as well. The data to be studied are all corrected to the particle level and includes only charged particles. Figure 6

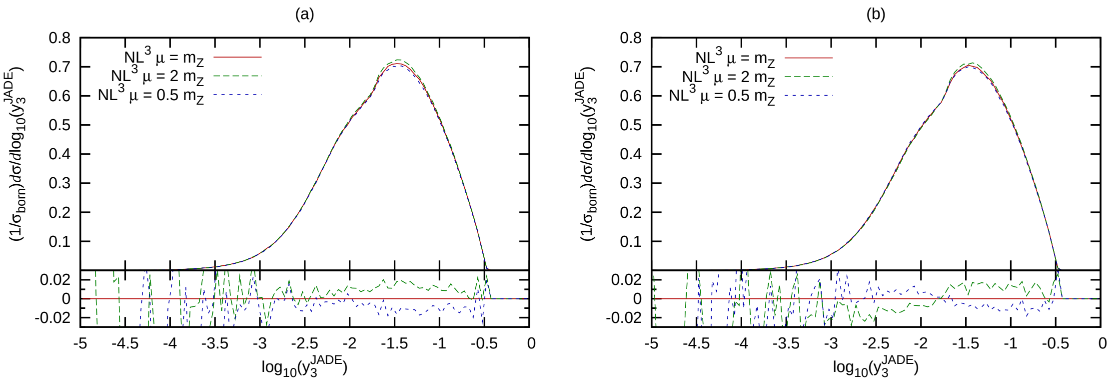


**Figure 5:** The parton-level $y_3$ spectra according to the JADE definition for the NL$^3$ algorithm including a modified way of treating $\alpha_s$ where the renormalization scale has been varied up and down by a factor of two. (a) includes the one-loop corrections and (b) includes the two-parton matrix element at two loops.

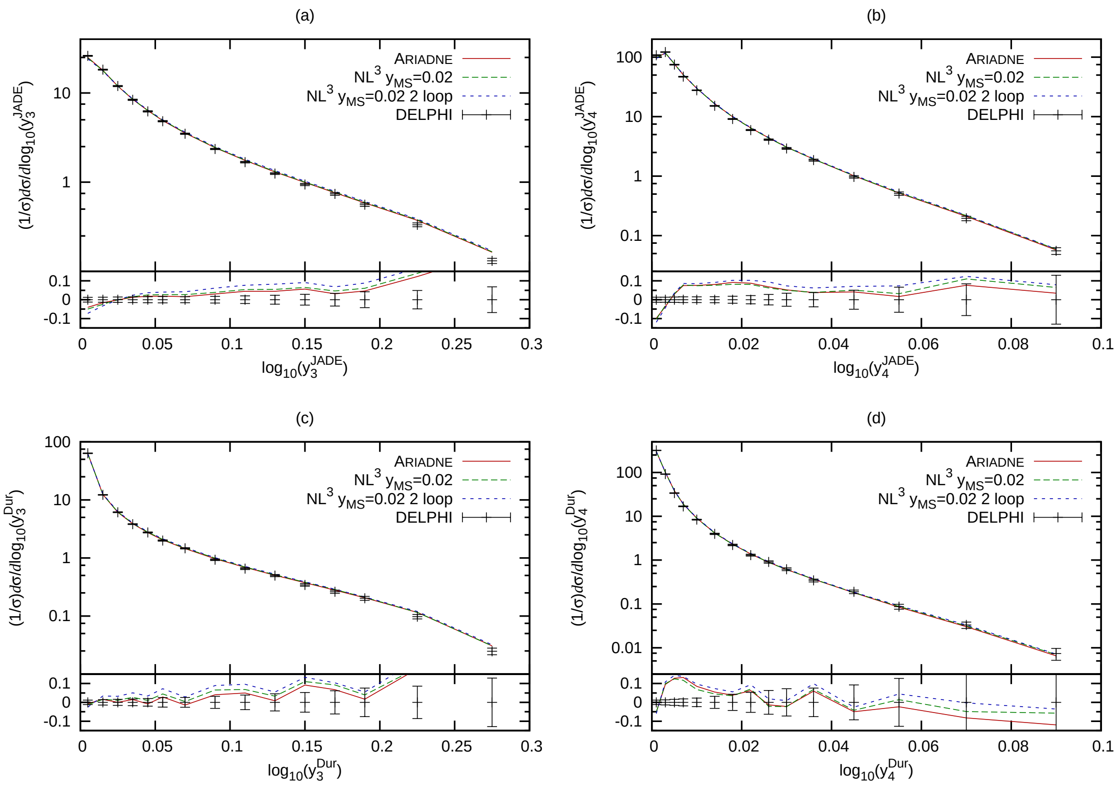


**Figure 6:** Charged particle jet observables compared to DELPHI data and the standard ARIADNE shower for the NL$^3$ algorithm using a merging scale of 0.02. The following jet observables are shown: (a) 3 jet JADE, (b) 4 jet JADE, (c) 3 jet Durham and (d) 4 jet Durham.

shows the jet distributions for the third and fourth jet according to the JADE [34] definition and the Durham [37] definition. The curves include the central merging scale value of 0.02

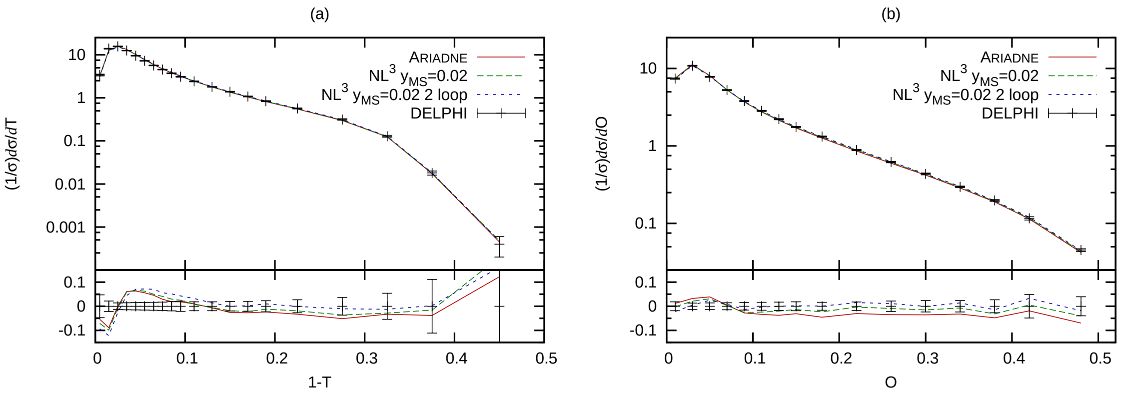


**Figure 7:** The charged particle thrust and oblateness compared to DELPHI data and the standard ARIADNE shower for the $NL^3$ algorithm using a merging scale of 0.02.

with and without the extra two-loop correction. All the results reproduce the data quite well.

The results from the algorithm has also been compared to the shape observables thrust and oblateness, which is shown in figure 7. The agreement is again quite good. The curves including the matrix element corrections actually do a bit better for oblateness, which is to be expected since it is sensitive to distributions including four jets where we have included the exact tree-level matrix elements.

## 5. Conclusions

We have presented an algorithm for merging one-loop and tree-level matrix elements with parton showers. The algorithm allows for the inclusion of several different multiplicities, which is important for simulating an entire process. For observables where the combination of tree-level and one-loop matrix elements used gives the correct NLO prediction, our procedure will also give correct NLO predictions but with a resummation of leading logarithms which is of the same accuracy as in the parton shower used. The basic principle of the procedure is quite simple. The first two terms in the $\alpha_s$ expansion is subtracted from the shower and the corresponding one-loop matrix element is added. Although a simple idea, it leads to some complicated issues.

To calculate the first two terms in powers of $\alpha_s$ in a parton shower requires that several different terms are taken into account. Both the first term in the expansion of the Sudakov form factor and the first term in the running coupling contributes at this level. In this paper these terms were derived and subtracted from the shower. The calculation was done within a framework similar to the CKKW-L method, but it is also applicable to CKKW algorithms in general.

The modified shower is then added to a sample calculated according to one-loop matrix elements. The requirement on the matrix elements is that one should be able to specify a phase-space cut which separates the parton multiplicities and decides whether or not a parton is to be considered unresolved. This is not done within the commonly used subtraction schemes, but can be solved by a modifying the subtraction scheme outside the singular regions.

We have explicitly calculated all the weights without taking into account the possibility of incoming hadrons, which is going to be the topic of a future publication. We then implemented the procedure and applied it to the process $e^+e^-$ to hadrons. The shower in ARIADNE was used and PYTHIA was used to generate the matrix elements.

To test the consistency of the algorithm, jet distributions at parton level were studied. The trivial case of two partons at one loop and three partons at tree level was found to have only small deviations, which disappeared if quark masses were excluded. The first nontrivial case was how the algorithm behaves for two partons at one loop, three partons at one loop and four partons at tree level. This led to a clear overshoot due to the fact that the terms in the matrix element is significantly bigger than those in the shower, which is compensated by using a much larger $\alpha_s$ in the shower compared to fits using matrix elements including loops. If the value of $\alpha_s$ is adjusted according to our prescription, the agreement is quite good.

One important aspect of our procedure is the ability to combine several different parton multiplicities in a consistent way. This allows us to obtain corrections to a NLO prediction for a given $n$-jet observable, stemming from cumulative sub-leading effects from the parton shower added to the $(n-1)$-jet states.

Other aspects of the algorithms was explored, such as including the two-parton matrix element to two-loop accuracy, which only led to slight changes below the merging scale. The sensitivity to the choice of renormalization scale was tested and a change of a factor of two in the renormalization scale results in changes of around two percent in the results.

The predictions of the algorithm has also been compared to four different jets observables and two shape observables measured at LEP. ARIADNE already provides a good description of the data and including the matrix element corrections gives similar agreement.

Overall the procedure has been shown to be consistent and give good results at hadron level. These relatively simple cases establish a proof of concept and a good staring point to explore the additional pieces needed to simulate processes with incoming hadrons. If the algorithm is developed further and the matrix element generators improved, then there are good prospects for being able to merge one-loop matrix elements and partons shower for the more interesting LHC processes.

## 6. Acknowledgments

We thank Torbjörn Sjöstrand for useful discussions. Work supported in part by the Marie Curie research training network "MCnet" (contract number MRTN-CT-2006-035606).

## References

[1] S. Catani, F. Krauss, R. Kuhn, and B. R. Webber, *JHEP* **11** (2001) 063, arXiv:hep-ph/0109231.

[2] F. Krauss, *JHEP* **08** (2002) 015, arXiv:hep-ph/0205283.

[3] L. Lönnblad, *JHEP* **05** (2002) 046, arXiv:hep-ph/0112284.

[4] N. Lavesson and L. Lönnblad, *JHEP* **07** (2005) 054, arXiv:hep-ph/0503293.

[5] M. Mangano, "The so-called MLM prescription for ME/PS matching." http://www-cpd.fnal.gov/personal/mrenna/tuning/nov2002/mlm.pdf. Talk presented at the Fermilab ME/MC Tuning Workshop, October 4, 2002.

[6] M. L. Mangano, M. Moretti, F. Piccinini, and M. Treccani, *JHEP* **01** (2007) 013, arXiv:hep-ph/0611129.

[7] S. Mrenna and P. Richardson, *JHEP* **05** (2004) 040, arXiv:hep-ph/0312274.

[8] J. Campbell and K. Ellis, "MCFM - Monte Carlo for FeMtobarn processes." http://mcfm.fnal.gov/.

[9] Z. Nagy, *Phys. Rev.* **D68** (2003) 094002, arXiv:hep-ph/0307268.

[10] C. F. Berger *et al.*, *Phys. Rev.* **D78** (2008) 036003, arXiv:0803.4180 [hep-ph].

[11] S. Frixione and B. R. Webber, *JHEP* **06** (2002) 029, arXiv:hep-ph/0204244.

[12] S. Frixione and B. R. Webber, arXiv:hep-ph/0612272.

[13] P. Nason, *JHEP* **11** (2004) 040, arXiv:hep-ph/0409146.

[14] S. Frixione, P. Nason, and C. Oleari, *JHEP* **11** (2007) 070, arXiv:0709.2092 [hep-ph].

[15] M. Kramer and D. E. Soper, *Phys. Rev.* **D69** (2004) 054019, arXiv:hep-ph/0306222.

[16] D. E. Soper, *Phys. Rev.* **D69** (2004) 054020, arXiv:hep-ph/0306268.

[17] M. Kramer, S. Mrenna, and D. E. Soper, *Phys. Rev.* **D73** (2006) 014022, arXiv:hep-ph/0509127.

[18] Z. Nagy and D. E. Soper, *JHEP* **09** (2007) 114, arXiv:0706.0017 [hep-ph].

[19] W. T. Giele, D. A. Kosower, and P. Z. Skands, arXiv:0707.3652 [hep-ph].

[20] C. W. Bauer, F. J. Tackmann, and J. Thaler, arXiv:0801.4026 [hep-ph].

[21] C. W. Bauer, F. J. Tackmann, and J. Thaler, arXiv:0801.4028 [hep-ph].

[22] L. Lönnblad, *Comput. Phys. Commun.* **71** (1992) 15–31.

[23] T. Becher and M. D. Schwartz, *JHEP* **07** (2008) 034, arXiv:0803.0342 [hep-ph].

[24] J. Alwall *et al.*, *JHEP* **09** (2007) 028, arXiv:0706.2334 [hep-ph].

[25] M. L. Mangano, M. Moretti, F. Piccinini, R. Pittau, and A. D. Polosa, *JHEP* **07** (2003) 001, arXiv:hep-ph/0206293.

[26] S. Catani and M. H. Seymour, *Phys. Lett.* **B378** (1996) 287–301, arXiv:hep-ph/9602277.

[27] S. Catani and M. H. Seymour, *Nucl. Phys.* **B485** (1997) 291–419, arXiv:hep-ph/9605323.

[28] P. M. Stevenson, *Phys. Rev.* **D23** (1981) 2916.

[29] N. Lavesson and L. Lönnblad, *JHEP* **04** (2008) 085, arXiv:0712.2966 [hep-ph].

[30] L. Lönnblad, *Comput. Phys. Commun.* **71** (1992) 15.

[31] T. Sjostrand, S. Mrenna, and P. Skands, *JHEP* **05** (2006) 026, arXiv:hep-ph/0603175.

[32] R. K. Ellis, D. A. Ross, and A. E. Terrano, *Nucl. Phys.* **B178** (1981) 421.

[33] R.-Y. Zhu, *Determination of the strong coupling constant up to complete second order in QCD.* Ph.d. thesis, MIT, 1983. MIT-LNS Report RX-1033.

[34] **JADE** Collaboration, W. Bartel *et al.*, *Z. Phys.* **C33** (1986) 23. See also S. Bethke, Habilitation thesis, LBL 50-208 (1987).

[35] **Particle Data Group** Collaboration, C. Amsler *et al.*, *Phys. Lett.* **B667** (2008) 1.

[36] **DELPHI** Collaboration, P. Abreu *et al.*, *Z. Phys.* **C73** (1996) 11–60.

[37] S. Catani, Y. L. Dokshitzer, M. Olsson, G. Turnock, and B. R. Webber, *Phys. Lett.* **B269** (1991) 432–438.