# An Analytic Parton Shower:

## Algorithms, Implementation and Validation

**Dissertation**

**zur Erlangung des Doktorgrades**

**des Fachbereichs Physik**

**der Universität Hamburg**

**vorgelegt von**

**Sebastian Schmidt**

**aus Lennestadt**

**Hamburg**

**2012**

| | |
|---|---|
| Gutachter der Dissertation: | Prof. Dr. Bernd Kniehl<br>Dr. Jürgen Reuter |
| Gutachter der Disputation: | Prof. Dr. Joachim Bartels<br>Dr. Jürgen Reuter |
| Datum der Disputation: | 11. Juni 2012 |
| Vorsitzender des Prüfungsausschusses: | Dr. Georg Steinbrück |
| Vorsitzender des Promotionsausschusses: | Prof. Dr. Peter Hauschildt |
| Dekan der Fakultät für Mathematik, Informatik und Naturwissenschaften: | Prof. Dr. Heinrich Graener |

**Abstract**

The realistic simulation of particle collisions is an indispensable tool to interpret the data measured at high-energy colliders, for example the now running Large Hadron Collider at CERN. These collisions at these colliders are usually simulated in the form of exclusive events.

This thesis focuses on the perturbative QCD part involved in the simulation of these events, particularly parton showers and the consistent combination of parton showers and matrix elements. We present an existing parton shower algorithm for emissions off final state partons along with some major improvements. Moreover, we present a new parton shower algorithm for emissions off incoming partons. The aim of these particular algorithms, called *analytic parton shower algorithms*, is to be able to calculate the probabilities for branchings and for whole events after the event has been generated. This allows a reweighting procedure to be applied after the events have been simulated.

We show a detailed description of the algorithms, their implementation and the interfaces to the event generator `WHIZARD`. Moreover we discuss the implementation of a MLM-type matching procedure and an interface to the shower and hadronization routines from `PYTHIA`. Finally, we compare several predictions by our implementation to experimental measurements at LEP, Tevatron and LHC, as well as to predictions obtained using `PYTHIA`.

**Zusammenfassung**

Die realistische Simulation von Teilchenkollisionen ist ein unverzichtbares Werkzeug um die an Teilchenbeschleunigern, zum Beispiel dem zur Zeit laufenden Large Hadron Collider (LHC), gemessenen Daten zu verstehen. Die an diesen Teilchenbeschleunigern erzeugten Kollisionen werden typischerweise in der Form von exklusiven Ereignissen simuliert.

Diese Arbeit konzentriert sich auf den Anteil perturbativer QCD an der Simulation dieser Ereignisse, im Besonderen Parton Showern und die konsistente Kombination von Parton Showern und Matrixelementen. Wir stellen einen bereits bekannten Parton Shower Algorithmus für Abstrahlungen von Teilchen im Endzustand harter Wechselwirkungen und unsere Verbesserungen an diesem Algorithmus vor. Des Weiteren zeigen wir einen neuen Parton Shower Algorithmus für Abstrahlungen im Anfangszustand. Das Ziel dieser Algorithmen, genannt *analytische Parton Shower Algorithmen*, ist, die Wahrscheinlichkeiten für die einzelnen Abstrahlungen und somit für komplette Ereignisse berechnen zu können, auch nachdem das komplette Ereignis generiert wurde. Dies erlaubt eine Neugewichtung der Ereignisse nachdem die Ereignisse simuliert wurden.

Wir präsentieren eine detaillierte Beschreibung der Algorithmen, ihre Implementierung und die Schnittstellen zum Ereignisgenerator `WHIZARD`. Außerdem besprechen wir die Implementierung einer Prozedur zur Kombination von Parton Showern und Matrixelementen, genannt MLM-matching, und eine Schnittstelle zu den Shower- und Hadronisierungs-Programmen aus `PYTHIA`. Zum Abschluss vergleichen wir verschiedene Vorhersagen unserer Implementierungen mit experimentellen Messungen von LEP, Tevatron und LHC, sowie mit Vorhersagen, die mit `PYTHIA` gemacht wurden.

# Contents

# List of Figures

# 1 Introduction

The current years are interesting times for particle physicists. The Large Hadron Collider (LHC), running since 2008, might give new insights and solve some problems of particle physics or even raise new open questions. To understand the measurements made at the two main-purpose detectors at the LHC, the ATLAS and the CMS detectors, predictive simulations of hadronic collisions need to be available. This thesis gives a description of my efforts to aid in the development of these simulations.

## 1.1 The Standard Model of Particle Physics

The *Standard Model of particle physics* (SM) has been very successful for the last decades as one of the cornerstones of physics. The model describes the elementary particles and their strong and electro-weak interactions. The particle content of the Standard Model consists of the quarks and leptons, the force-carrying gauge bosons and the Higgs boson, that via its non-vanishing vacuum expectation value generates the masses of other elementary particles. The quarks and the gluon, one of the force-carriers, are of particular interest as these are the particles carrying a color charge. The associated interaction shows two peculiar features: *confinement* and *asymptotic freedom*. Confinement explains that quarks and gluons are only found in color-neutral bound states, while asymptotic freedom means that quarks and gluons appear to be free fields at higher energies. The tension between these two features is one underlying motif of this thesis.

In the past, the Standard Model has been successfully validated many times. Some of the most important experimental confirmations are the discoveries of weak neutral currents [2], the *charm* quark [3, 4], the $W$- and $Z$-Bosons [5, 6], the *top* quark [7, 8] and the $\tau$ neutrino [9], as well as the confirmation of the predictions for the anomalous magnetic dipole moment of the muon, $g-2$, and the agreement between values for the weak mixing angle $\theta_W$ measured in a variety of ways, all verified by experiments with outstanding precision [10]. However, the Standard Model awaits further confirmation as the search for the Higgs boson has not been successful as so far. However, recent data seem to suggest the existence of the Higgs boson. The search for this key ingredient of the Standard Model was one of the main reasons to build the LHC and it is very probable that either the discovery or the exclusion of the Higgs boson will be announced within this year.

The Standard Model is formulated in the framework of relativistic quantum field theory. Analytic calculations are only feasible for simple observables, and even then, a direct comparison of these observables to measure-

ments is difficult. In general, more complicated methods have to be used. Four paths of predicting observables in collider physics are widely pursued:

- Perturbative calculations.
- Analytical resummation methods.
- Lattice calculations.
- Phenomenological models for non-perturbative physics.

The perturbative ansatz is the ansatz that we will pursue in this thesis, however supplemented with models for non-perturbative physics where necessary. The aim of resummation methods is to describe inclusive quantities by analytically resumming higher-order contributions. Resummation can be applied to collider physics, see e.g. [11], but is limited to specific observables. The ansatz of lattice calculations is to discretize space-time into a lattice and then solve the problem numerically. Lattice calculations have been successfully applied to stationary processes, like the calculation of hadron masses [12], but are too computationally demanding to simulate scatterings at a collider. Models for non-perturbative physics are used to describe hadron-hadron interactions with low momentum transfer, like elastic scatterings, diffractive scatterings or low $p_T$ non-diffractive events. These scatterings can be modelled by the use of pomeron and reggeon exchange. See for example [13] for a review of these soft interactions at the LHC.

In the perturbative ansatz, observables are given in a power expansion in the strong coupling constant $\alpha_S$[1],

$$\mathcal{O} = \sum_{i=1}^{\infty} \mathcal{O}_i \alpha_S^i, \tag{1.1}$$

which in principle calls for the calculation of infinitely many contributions. It is therefore preferable to choose observables that are only weakly affected by higher-order contributions. If $\alpha_S$ is small, a sufficient approximation can be obtained by taking only a limited number of orders into account[2],

$$\mathcal{O}_n = \sum_{i=1}^{n} \mathcal{O}_i \alpha_S^i. \tag{1.2}$$

Therefore, every perturbative calculation is inherently dependent on the maximal number of powers of $\alpha_S$ taken into account. Thus, every perturbative

[1] Although we only use $\alpha_S$ the same applies for electro-weak interactions and analogously for combinations of both.

[2] assuming an insignificant volatility of the coefficients $\mathcal{O}_i$

calculation is afflicted with the error of omitting higher orders. Moreover, as the complexity to calculate the coefficients $\mathcal{O}_i$ grows rapidly with order $i$, many observables have only been calculated up to the first non-vanishing $\mathcal{O}_i$, hence being called *leading order* (LO) calculations. Due to theoretical effort in the last years, more and more *next-to-leading order* (NLO) calculations and generators become available[3], while calculations at *next-to-next-to-leading order* (NNLO) are very scarce.

The most important observables are the cross sections for specific processes, that can be directly related to the probability for each process. However, calculated naively, cross sections show divergences that can stem from a variety of sources. For this thesis, soft and collinear divergences are of particular interest. Soft divergences are caused by massless particles having an energy close to zero, while collinear divergences are caused by two particles moving in directions very close to each other. These divergences do not reflect physical divergences, they only appear as a consequence of an inappropriate definition of observables. For observables defined suitably inclusive it has been shown by Bloch and Nordsieck [17] and Kinoshita, Lee and Nauenberg [18, 19] that these divergences cancel (for a more recent review see [20]).

## 1.2 Experimental Procedure

In the early days of particle physics, the analysis of an experiment consisted of screening cloud chamber or bubble chamber photographs and analyzing the visible tracks. A track that could not be attributed to any known particle could represent the observation, in the true sense of the word, of an unknown particle. One of the discoveries made this way is the discovery of the positron in 1933 [21], where 15 out of 1300 photographs of a cloud chamber displayed an unknown particle that we today know as the positron.

During the last decades, physicists have been gaining more and more knowledge about the elementary particles and hence pushed the frontier of ignorance further and further. The driving force was the development and running of colliders with continuously increasing energies, accompanied by detectors with ever increasing resolution. These high energy experiments have led to many of the discoveries mentioned in the previous chapter and made the term *high energy physics* a synonym for particle physics. The latest collider in this line is the LHC with its two multi-purpose detectors ATLAS and LHC.

[3] For a review of NLO calculations see [14], for a review of NLO generators see [15], or the more general recent review [16].

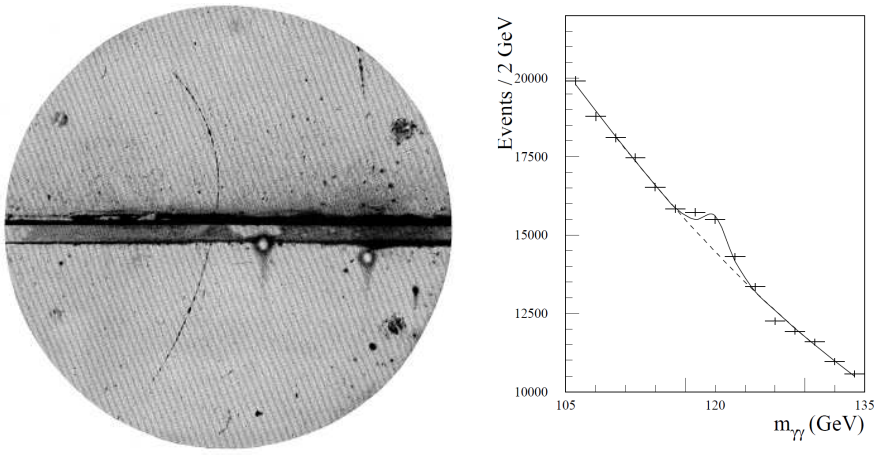


Figure 1.1: Pushing the frontier of ignorance during the last 80 years: (left) One of 1300 cloud chamber images from the experiment that led to the discovery of the positron [21]. The track shown could not be identified with any known particle, the most likely explanation was that an until then unknown particle has been observed. (right) A histogram for the distribution of the mass of photon pairs, obtained from a simulated measurement at the ATLAS detector [22]. The expected distribution assuming the non-existence of the Higgs boson is shown as the dashed curve, while the solid curve shows the expected distribution assuming the existence of a Higgs boson with a mass $m_H = 120\,\text{GeV}$. A simulated measured distribution is shown as the crosses, this distribution obviously favors the existence of a Higgs boson over the non-existence.

As of 2011, the LHC collides protons at a center-of-mass energy of 7 TeV. In the individual events, some of the protons' constituents, quarks and gluons, collide and produce a multitude of final states. Produced photons, electrons and muons can be directly measured in the detector. On the contrary, quarks and gluons cannot be measured directly. Due to color confinement they will form hadrons that can be measured. These hadrons usually appear in the form of bunches traveling in a similar direction. These bunches are called *jets*[4].

It is assumed that all particles measured in the detectors are known, while any unknown particle will either decay before reaching the detector or escape the detector without leaving any signal. Thus the only way to gain knowledge is to analyze the measured particles and try to deduce information about the intermediate states from them.

The modern way of analyzing the measurements is to simulate a large number of events and compare histograms of observables, like e.g. missing transverse momentum, obtained from the simulated events and compare them with the corresponding histograms obtained from real measurements. If a satisfactory degree of consistency is present, the theory can be regarded to describe the data in a correct way, while a significant deviation shows the existence of physics unaccounted for, either a deficiency in the simulation or possibly new physics.

## 1.3 Event Generators

With the need for simulated event samples, the demand for *event generators* rose. Event generators are computer programs that transform theoretical input into simulated events. These events can be either weighted or unweighted. Unweighted events each contribute with equal weight to any analysis, while the contribution of weighted events is given by their respective weight. An event generator usually generates weighted events in the first place, unweighted events are generated by rejecting weighted events according to their weight. Only unweighted events can be directly compared to measured data. However, measurable observables are calculated using these events. These observables are mostly either cross sections for specific processes. The simulation of events at a present-day collider is demanding for a combination of various physical aspects. Modern event generators thus divide the generation of an event in various steps, see the next section for a more detailed discussion of these steps.

Nowadays, several multi-purpose event generators are publicly available.

[4] See section A.2 for one way of defining jets.

Probably the event generator most used today is PYTHIA [23, 24], with ALPGEN [25], Herwig++ [26], MadGraph [27], Sherpa [28] and WHIZARD [29] as widely used alternatives. Moreover, there exists a plethora of programs that are tailored to one special purpose and are mostly designed to work as extensions or plug-ins to the more common event generators. For example LHAPDF [30] provides a standardized interface to parton distribution functions and FeynRules [31] calculates the set of Feynman rules associated with a given Lagrangian and allows these rules to be used in the event generator.

## 1.4 Typical Generation of an Event

In modern event generators, the generation of events is divided into several distinct steps. For each step, the simulation is performed by a dedicated component of the generator. A precise simulation of an event therefore demands a coordinated interplay of these components. The needed steps are the calculation of hard matrix elements, the description of the hadronic substructure by parton distribution functions, the simulation of parton showers, a matching procedure to combine matrix elements and parton shower, the modeling of multiple interactions, the hadronization, the hadronic decays and the simulation of the detector response.

The steps from a hard interaction to a fully simulated event at a hadron-hadron collider are shown in the following. For an event at a lepton-lepton collider, some steps can be omitted or simplified. The simulation can be divided into four major steps: the hard interaction, the residual hadron-hadron interaction, pile-up and the detector simulation.

### 1.4.1 The Hardest Interaction

The *hardest* or *core interaction* is the scattering with the highest energy scale or momentum transfer. It is a scattering of two incoming particles or partons into $n$ outgoing partons. The calculation of processes with a high number of outgoing partons $n$ is limited by the rapid growth of complexity of the corresponding matrix elements and the integration over the phase space. On present-day computers $n$ is bound to be only up to six to eight for leading-order calculations. The hardest interaction usually incorporates the most interesting physical process of the event. For example, at the LHC it might be the production of a Higgs boson and its subsequent decay. The hardest interaction is described by means of a perturbative series in the framework of quantum field theory. Mostly, except for the omission of higher order terms in the perturbative expansion, no approximations are made in the matrix elements used to generate the hardest interaction.

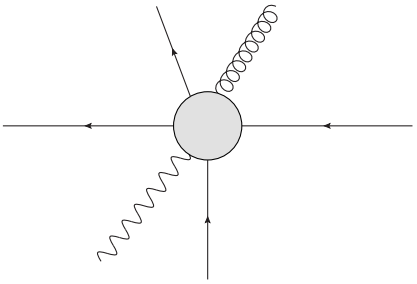

Figure 1.2: Generating an event: the hard interaction, for example a $2 \to 4$ process with two incoming quarks, two outgoing quarks, one outgoing gluon and one outgoing photon. The internal structure of the hard interaction is not shown, it is just represented by the gray circle.

The component of an event generator that simulates the hardest interaction is generally the most important component of the event generator.

### 1.4.2 The Residual Hadron-hadron Interaction

Here, we refer to everything else that occurs associated with the hard interaction as the *residual hadron-hadron interaction*.

**Parton Distribution Functions** When colliding hadrons, the fact that hadrons are composite objects and that it is not the hadrons but their components that actually collide has to be taken into account. The colliding constituents of the hadrons have been named partons. From today's point of view, one knows that these partons can be quarks or gluons and due to the *factorization theorem*, the probability density for a parton to be part in a collision can be described by *parton distribution functions* (PDFs). The hadron is thought to be in a superposition of all possible compositions of partons. The PDF for a specific type with a given momentum fraction is then proportional to the fraction of possible compositions that contain the specified parton with the specified momentum fraction. The parton distribution functions then give the probability density to find a parton of certain type with momentum fraction $x$ in a hadron at a scale $Q^2$ taking part in a hard interaction. The cross section for the production of a state $X$ is then given by

$$\sigma_X = \sum_{i,j} \int \mathrm{d}x_1 \mathrm{d}x_2 f_i\left(x_1, Q^2\right) f_j\left(x_2, Q^2\right) \hat{\sigma}_{ij\to X}\left(p_1, p_2, Q^2\right) \tag{1.3}$$

with the partonic cross section $\hat{\sigma}_{ij\to X}$ and the PDFs $f_i$ and $f_j$.

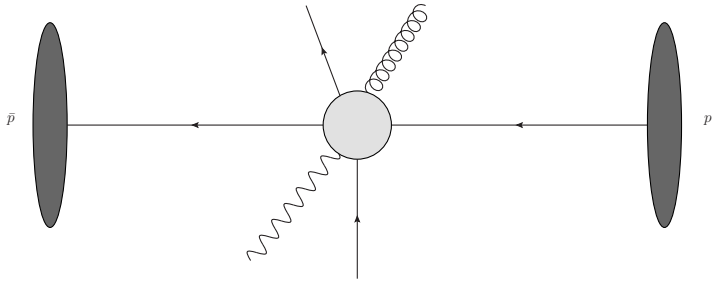


Figure 1.3: Generating an event: adding parton distribution functions. The hadrons are shown as dark ellipses. In this case, an antiproton is shown on the left, a proton is shown on the right.

As hadrons are nonperturbatively bound objects, PDFs cannot be calculated in the framework of perturbative quantum field theories. The only other way to determine PDFs from first principles is via the use of lattice QCD calculations, but these have not yet proven successful. Therefore PDFs can only be obtained by fitting experimental results and are thus always tainted with experimental, statistical and fitting-induced uncertainties. Various collaborations obtain PDFs using different ways of fitting and including different data sets. The most well-established PDFs are those published by the CT10 [32], MSTW [33] and NNPDF [34] collaborations.

**Parton Showers** *Parton showers* describe additional emissions off the incoming and outgoing partons in an interaction. The parton shower consists of the *initial-state parton shower* or *initial-state radiation* (ISR) describing emissions off the incoming partons and the *final-state parton shower* or *final-state radiation* (FSR) describing emissions off the outgoing partons.

The necessity for parton showers can be seen when comparing a process with charged external particles to the same process with an additional parton emitted, e.g. comparing $e^+e^- \to q\bar{q}$ to the same process with one additional gluon, $e^+e^- \to q\bar{q}g$. For a small minimal gluon energy or a small minimal emission angle between the gluon and one of quarks, the ratio of the cross sections $\sigma(e^+e^- \to q\bar{q}g)/\sigma(e^+e^- \to q\bar{q})$ will diverge to infinity. This breakdown of perturbation theory is due to the incorrect assumption that the particles in the final state are distinct particles. On closer examination every charged particle is inevitably surrounded by a cloud of virtual gauge bosons. The divergent contributions stem from processes where the additional parton is

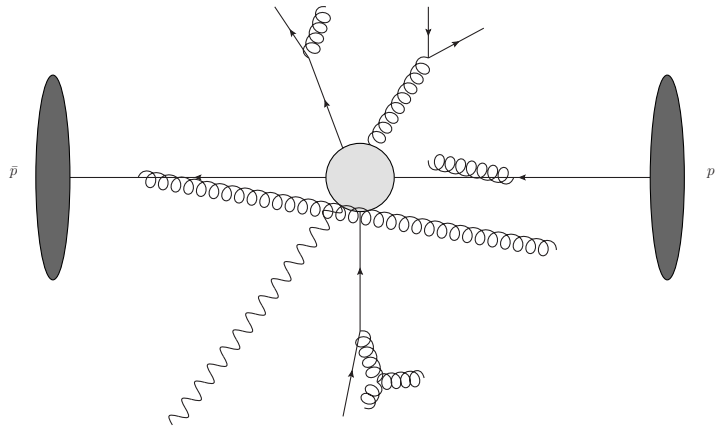


Figure 1.4: Generating an event: adding parton showers. The incoming and outgoing quarks emit gluons. One of these gluons splits further into two gluons. The gluon produced in the hard interaction splits into a quark-antiquark pair.

part of another parton's cloud, but is treated as an individual particle. However, extrapolating the divergent contributions away from the divergence, observable effects can be obtained, manifesting themselves in the emission of observable partons, which is then called the parton shower. These partons might initiate additional jets or influence the substructure of the jet they are produced in.

The framework of parton showers is a well defined approximation to the full matrix element. The most divergent contributions take the form $\int \frac{\mathrm{d}t}{t}$ and hence correspond to logarithmic divergences. These contributions are thus called *leading logarithms* (LL), while the inclusion of higher order logarithms is possible. The implementation is in the form of a Markov chain of parton branchings, where all branchings are treated independently.

**Matching** The matrix element describes processes with a given number of external partons up to a given order in the coupling constant, giving a correct description of hard emissions. On the other hand, parton showers describe events with a variable number of additional particles by resumming leading logarithms, giving a correct description of soft-collinear emissions. Thus these two approaches describe different kinds of emissions correctly and it is therefore preferable to combine a description of emissions using matrix elements with a parton shower description where they are applicable. This leads to *matching* procedures that aim to combine these approaches consistently.

For this, it is necessary to avoid possible double-counting of contributions and ensure a smooth transition between the two ways of description. Figure 1.5 gives a pictorial description of the combination of matrix elements and parton showers.

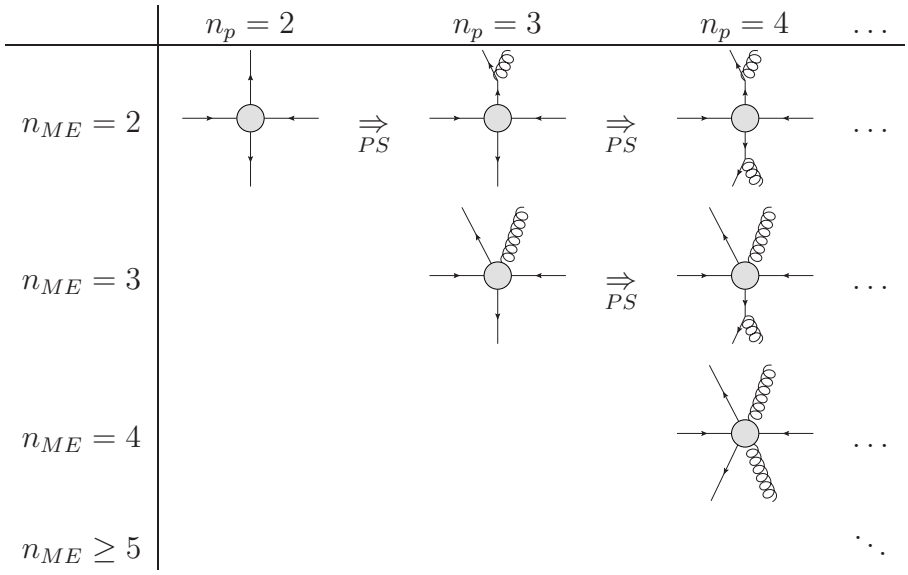


Figure 1.5: Generating an event: matching. The interactions described by matrix elements are shown as gray circles. For the parton shower emissions, only one example emission is shown here. The aim of matrix element/parton shower matching is to combine the descriptions shown in each column. For example, an event with four final partons, $n_p = 4$, could stem from a $2 \to 2$ matrix element followed by two emissions during the parton shower (first row), a $2 \to 3$ matrix element followed by one parton shower branching (second row) or directly from a $2 \to 4$ matrix element (third row).

**Multiple Interactions** As hadrons are compound objects, the possibility that more than one pair of components of each hadron interact cannot be neglected. In principle, due to the rise of PDFs at low momentum fraction $x$, arbitrarily many parton-parton interactions could occur in each hadron-hadron interaction. The number of interactions is limited due to the fact that most of these interactions would be soft and thus the partons would not be able to resolve the respective other hadron's internal substructure. With reasonable cuts the average amount of parton-parton interactions per

proton-proton collision at the LHC is of the order of 10. All aspects already mentioned apply to additional interactions as well: the interactions have to be weighted by PDFs themselves, where the PDFs have to be adjusted for subsequent interactions as the former interactions have altered the flavor, color and momentum composition of the hadron or the hadron remnant. Moreover, all particles from the additional interactions can start parton showers on their own. Matching can be applied as well, but is usually not necessary as the additional interactions are rather soft and thus probably incapable to produce additional hard emissions.

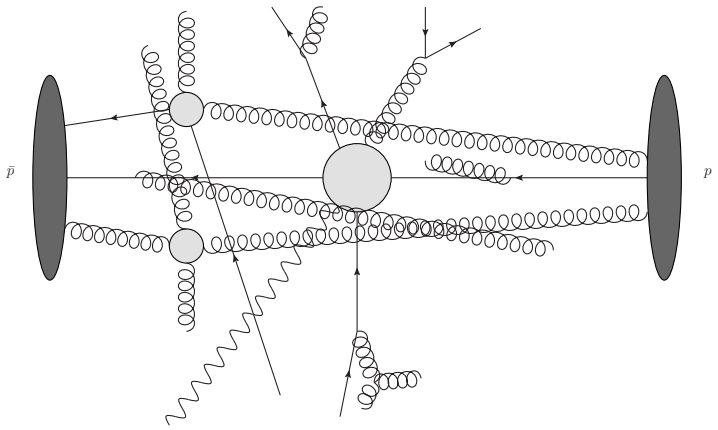


Figure 1.6: Generating an event: multiple interactions. In this case two additional interactions are shown as gray circles, one $gg \to gg$ and one $\bar{q}g \to \bar{q}g$ interactions.

This approach relies on the assumption that the interactions proceed independent of each other. However, the interactions can be connected by sharing external particles. These connected interactions can be divided into two classes, *joined interactions* and *rescattering*.

In the first special case of joined interactions, a branching in the ISR of one hadron produces two partons that interact with two partons from the other hadron, thereby letting a $2 \to n_1$ and a $2 \to n_2$ effectively mimic a $3 \to n_1{+}n_2$ interaction. In the case of rescattering, a parton from the final state of one interaction scatters again against another parton from one of the hadrons in a subsequent interaction, thus connecting the final state of one interaction with the initial state of another interaction. Rescattering occurring in an event with one $2 \to n_1$ interaction and one $2 \to n_2$ interaction will mimic a $3 \to n_1 + n_2 - 1$ interaction. Rescattering involving two $2 \to 2$ interactions could therefore be visible in three-jet-rates. This has been studied [35] but

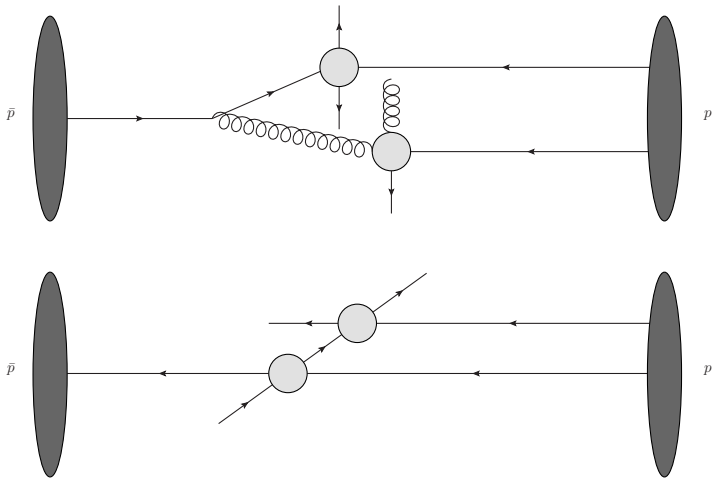


Figure 1.7: Generating an event: multiple interactions with joined interactions (top) and rescattering (bottom).

not found to be of significant importance.

**Interleaved Evolution** The simulation of multiple interactions, initial-state radiation and PDFs has to be performed in a combined way. This is necessitated by the need to distribute the hadron's flavor content and momentum among the particles produced in the interactions and the initial-state parton showers. To prohibit a violation of conservation of quantum numbers and momentum, the algorithms simulating the parton shower and multiple interactions must take the full set of resolved branchings and multiple interactions into account. One approach for this is the *interleaved evolution* [36]. The basic idea is to combine the simulation of multiple interactions and initial-state parton showers in one Markov chain, where at each step, either an additional ISR branching or an additional multiple interaction is found. During this Markov chain, parton distribution functions that have been adjusted for all interactions and all branchings in the initial-state radiation so far have to be used. Developing an algorithm to calculate these adapted PDFs is a non-trivial task [37].

A different approach to the steps denoted here as *multiple interactions* and *interleaved evolution* is the use of multi-parton distribution functions. Currently, only double-parton distribution functions are available [38]. Then

equation (1.3) will be expanded to look like

$$\sigma_{XY} = \frac{m}{2\sigma_{eff}} \sum_{i,j,k,l} \int \mathrm{d}x_1 \, \mathrm{d}x_2 \, \mathrm{d}x_3 \, \mathrm{d}x_4 \, D_{ij}\left(x_1, x_2, \mu_X^2, \mu_Y^2\right) D_{kl}\left(x_3, x_4, \mu_X^2, \mu_Y^2\right) \hat{\sigma}_{ik\to X}\left(p_1, p_3, \mu_X^2\right) \hat{\sigma}_{jl\to Y}\left(p_2, p_4, \mu_Y^2\right) \quad (1.4)$$

with the symmetry factor $m$ and the normalization $\sigma_{eff}$. The double-parton distribution functions have to be obtained from data, which has been found to be more difficult than for single-PDFs. Also all limitations from single-PDFs apply for double-PDFs and multi-PDFs as well.

**Simulation of the Beam Remnant** After initial-state radiation and multiple interactions have been simulated, the hadron remnant will carry remaining momentum and the remaining flavor and color quantum numbers. As the remnant is most likely to move near the beam axis it will hardly be seen in the detector. Nevertheless, conservation of global momenta and quantum numbers has to be taken into account in order to provide a consistent state as input to the hadronization. The beam remnant will be replaced by a composition of gluons, quarks, diquarks, mesons and baryons with the correct color and flavor quantum numbers and the remaining momentum distributed among them. The actual composition of the beam remnant has to be chosen according to a specific model, as the corresponding physics is nonperturbative, see for example [23, chapter 11] for the models implemented in `PYTHIA`.

**Hadronization** Due to confinement, colored particles combine to form hadrons before reaching the detector. This *hadronization* is a nonperturbative process and hence can only be described by empirical models and not from first principles. Hadronization can be regarded to be the inverse of the transition from hadrons to partons using PDFs. The mostly used implementation of a hadronization model is the implementation of the *Lund string model* [39] in `PYTHIA`. Another popular hadronization model is the *cluster model* [40] with its implementation in Herwig++.

In the Lund string model it is assumed that the event is evaluated in the limit where interference terms between different color configurations are neglected, so that every particle is associated with one, or if the particle is a gluon two color partners. From these mutually connected color partners connected strings can be composed. These consist of a color charge, mostly a quark, at the beginning, gluons along the string and an anticolor charge, mostly an antiquark, at the end of the string. Now additional $q\bar{q}$-pairs are created, either by cutting a color string or replacing a gluon. This resembles

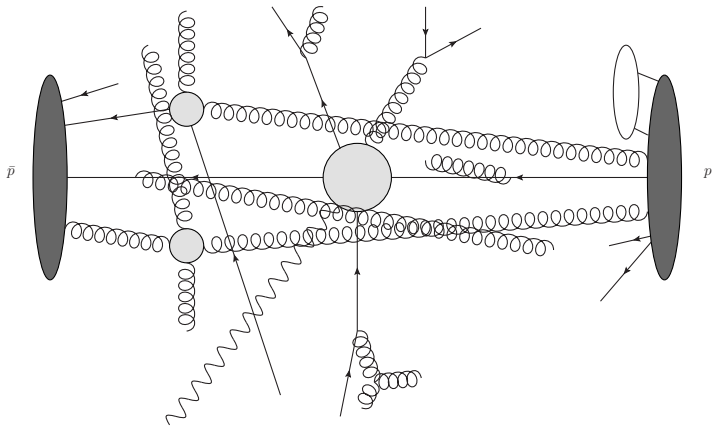


Figure 1.8: Generating an event: adding beam remnants. The antiproton's beam remnant consist only of a quark, the proton's beam remnant consists of a meson (shown as a white oval) and two more quarks.

the creation of additional quark-antiquark pairs from the color field when quarks are pulled apart. This procedure is iterated until all strings are small and accordingly low energetic enough. Each remaining string is then assumed to correspond to a meson and is replaced by this meson. The production of baryons can be implemented in this model by allowing diquark-pairs to be created along the string, so that remaining strings will correspond to three quark states.

Hadronization models typically employ a large number of model parameters. Dedicated studies are performed in order to determine optimal values for these parameters as they cannot be inferred from theory alone. These sets of optimal values are commonly referred to as *hadronization tunes*.

A different approach to hadronization is the use of *fragmentation functions*. These functions describe the probability for a quark or gluon to convert into a hadron of specific type [41, 10, chapter 17]. The fragmentation functions can be thought of as being the inverse of the parton distribution functions.

**Hadronic Decays** Some of the hadrons produced in the hadronization will decay too rapidly to reach the detector. Moreover, several of these decays can occur sequentially until all particles are stable enough to reach the detector. These decays can be simulated by incorporating the decay tables from the decay tables collected by the Particle Data Group (PDG) [10]. However,

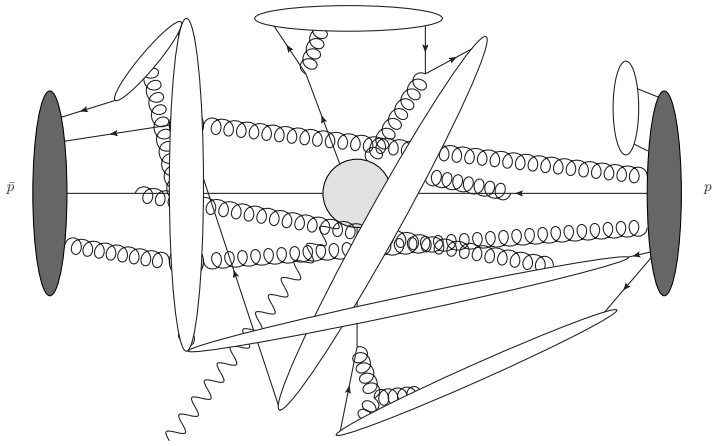


Figure 1.9: Generating an event: adding hadronization. The hadrons are shown as white ovals.

many decays are not precisely measured, the implementation therefore has to estimate the properties for these decays.

### 1.4.3 Pile-Up

Until now, only one interaction of one hadron from the first beam and one hadron from the second beam has been considered. As in the experiment the beams do not consist of single hadrons but of bunches of hadrons, about $10^{11}$ protons for the LHC, more than one hadron-hadron interaction per bunch-crossing will be possible. As of the end of 2011, the ATLAS experiment reported a mean number of interactions per crossing of 11.6[5], expecting an increase of this number for the near future. These multiple hadron-hadron interactions are called *pile-up*. However, it has to be noted, that the better part of these interactions is soft due to the relatively small cross section for hard interactions. The majority of bunch crossings containing at least one hard interaction will contain just one hard interaction. Thus, pile-up can be simulated by overlaying a hard hadron-hadron interaction with several soft hadron-hadron interactions as it is reasonable to assume that these interactions happen independent of each other.

[5]See section D for the ATLAS results.

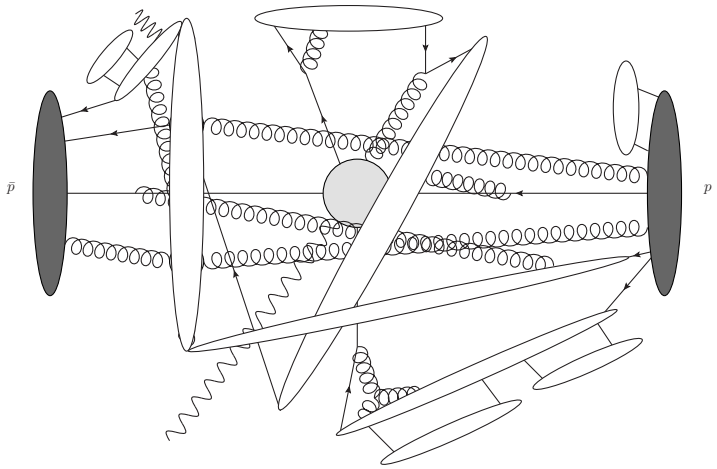


Figure 1.10: Generating an event: adding hadronic decays. Two of the hadrons decay, one into two hadrons, the other one into a hadron and a photon.

### 1.4.4 Detector Simulation

Finally, for comparison of the generated event sample with a measured event sample, a simulation of the detector itself is required. Detector deficiencies like dead zones, limited resolution and non-maximal trigger efficiency have to be taken into account. Moreover uncertainties can be introduced, for example due to unknown detector parameters or non-perfect calibration. Big efforts are made in order to perform reliable simulations of detectors, see e.g. for ATLAS [42] and for CMS [43].

After the detector simulation, the events can be analyzed and compared to the analyzed measured data. The detector simulation is beyond the scope of this thesis and will not be considered in the following. However, the detector simulation is the most time-consuming part of generating an event sample. For a typical Standard Model event, the time needed to simulate the ATLAS detector for one event is approximately 15 minutes [42, section 3.2].

### 1.4.5 Sequence of Steps

The procedure as given in sections 1.4.1 to 1.4.4 is not ordered corresponding to causality, but in a sequence going from high-scale to low-scale physics. This sequence is the sequence used during the generation of an event. An ordering according to causality would be possible as well, placing the par-

ton distribution functions and the initial-state radiation first, leading to the hardest interaction and the additional multiple interactions, then followed by final-state radiation, hadronization, decays and the detector simulation. This procedure would have the drawback that the parton distribution functions and the initial-state radiation would create a plethora of states unsuitable for the hard interaction to be simulated, as for example the flavors of the particle or the partonic center of mass energy would prohibit the hard interaction investigated to be produced. Therefore, an algorithm producing events in this way would be forced to reject a sizable part of the generated events, wasting a large fraction of computing time.

## 1.5 The Benefits of Analytic Parton Showers

As outlined in the previous section, parton showers are an indispensable part in the simulation of collisions in high energy physics.

Along with the implementation of multi-purpose event generators, many implementations of parton showers were developed. The parton shower implementations most commonly used are those in `PYTHIA` [23, 24], that are also invoked by many other programs using the provided interfaces. Alternatively, programs such as Herwig++ [26] and `SHERPA` [28] contain implementations of their own parton shower algorithms as components for their event generation frameworks. Moreover programs that are dedicated exclusively to parton showers which work as plug-ins to event generators are available, the most prominent being Ariadne [44] and Vincia [45].

These implementations have one common drawback: for a generated event it is not possible to calculate the probability for this event to be generated. This is due to the fact that the four-momentum conservation is enforced in every parton splitting in the shower, while the parton shower is implemented in a way that also generates splittings that do not conserve momentum. These splittings are then either dismissed or manually altered. But it is these splittings that modify the probability distributions in a practically incomprehensible way, making it impossible to calculate the branching probabilities for a particular event. If the probabilities were calculable, the effect of a change in the configuration of the parton shower could be included by calculating the branching probabilities and apply a weight corresponding to the probability before and after the change to each event. The only way to obtain an event sample according to changed settings would be to regenerate the complete event sample. This is unfavorable as all steps from the event generation, except for the generation of the hard interaction which is insensitive to the parton shower, would have to be repeated, including the most time-consuming part of simulating the detector response.

Recently it was shown that the formulation of parton showers can be reproduced by the *soft-collinear effective field theory* (SCET) [46]. This led to the formulation of so-called *analytic parton showers* [47]. Particular interest was placed on the the capability to calculate the probability for the branchings analytically, hence calling these parton showers "analytic". The advantage of having an analytic parton shower is the calculability of the probabilities, giving the potential to reweight events by the ratio of probabilities when changing the parton shower.

When using an analytic parton shower, changing the configuration of the parton shower program would only call for a resimulation of the parton shower. This resimulation would give a new weight for each event, corresponding to the ratio of the probabilities for generating the event using the old and new settings respectively. By this, one avoids the need to repeat the complete event simulation.

In this thesis, we intend to present an analytic parton shower for event generation. We extend the analytic final-state parton shower presented by Bauer et al. [47, 48] and develop a new analytic parton shower for initial-state radiation [49]. The development of this analytic initial-state parton shower, its implementation and its validation are the main topics of this thesis.

Different approaches to the same problem have been published in the literature. Stephens and van Hameren [50] presented an approach using variational techniques. A different approach was presented by Giele, Kosower and Skands [51] recently. They use an algorithm where trial emissions are generated using an overestimate of the splitting probability, a fraction of the trial splittings is then vetoed to regain the original splitting probabilities. The vetoed branchings are stored nevertheless. The event's weight under a different scheme can then be calculated, not analytically, but in a probabilistic way. It is given by the ratio of the joint probabilities to accept all accepted branchings while vetoing all vetoed branchings.

## 1.6 Structure of this Thesis

Let us shortly summarize the structure of this thesis: Chapter 2 contains an introduction to the theory of parton showers as well as an overview of the main approaches to the problem of combining matrix elements and parton showers, known as matching procedures. In section 3 we start by shortly summarizing the existing approach for an analytic final-state parton shower. In the main part of the section, we will then present our newly developed extension for the analytic final-state parton shower and the newly developed analytic initial-state parton shower. In section 4 we will describe technical details about the implementation of the analytic parton shower algorithms

within the event generator WHIZARD. Results obtained using this implementation are presented in section 5 along with comparisons of these results to data measured at LEP, Tevatron and LHC as well as to results obtained using the parton shower implemented in the event generator PYTHIA. Finally we will present our conclusion and an outlook for future plans connected with the analytic parton shower and its implementation.

The appendices contain supporting material for this work: In appendix A we describe the observables and jet measures used in the description of events. Appendix B contains three calculations that were too technical for the main text. We present sample source code in appendix C: One main program written in Fortran as a sample standalone program in section C.1 and four sample input files for WHIZARD in section C.2. Finally, appendix D shows an ATLAS measurement of the multiplicity of hadron-hadron interactions.

# 2 Theory of Parton Showers and Matching

Complete perturbative calculations in QCD can only be performed with a limited and fixed number of internal and external particles. Moreover the number of vertices and thus coupling constants is also limited, primarily due to the roughly factorial growth of complexity with increasing order of the coupling constants. Although this can be reduced to exponential growth by means of recursive algorithms like ALPHA [54], `O'Mega` [55] or Helac [56, 57], this is still not feasible for interactions with more than about ten particles in the final state, especially as the phase space integration becomes a major bottleneck for these processes. However, in some regions of phase space, contributions from higher orders and thus corresponding to a higher and possibly unlimited number of particles can be enhanced and can therefore not be neglected. These higher order contributions are enhanced by logarithmically divergent factors for each order of the coupling constant. For regions with enhancement due to soft or collinear particles, the approximation technique called *parton shower* has been developed.

An example of a phase space region where higher orders are enhanced is the annihilation of an $e^+e^-$ pair, creating a quark-antiquark pair, followed by the emission of nearly collinear gluons. The shape of these pencil-like events can be described by an event-shape variable called *thrust*[6] $T$, where $T \to 1$ for a 2-jet event with highly collimated jets. The cross section in these regions of phase space can still be expanded in orders of the coupling constant $\alpha_S$, but every contribution of $\alpha_S^n$ will be accompanied by a factor of the form $\ln^{2n}(1-T)$. It is this additional factor that renders the fixed-order calculations unreliable in the phase space region of highly collimated 2-jet events, corresponding to $T \to 1$.

Two main approaches exist on how to handle these regions of phase space: the first ansatz is to use a *resummation calculation*[11]. In this ansatz it is tried to resum the divergent contributions to gain a meaningful and convergent result. However, the resummation ansatz has considerable drawbacks as it can only be used to calculate inclusive observables and has to be adapted for every new observable. Moreover, resummation does not give a full representation of events which makes it impossible to incorporate detector effects in a complete experimental analysis. A more general ansatz is the use of a parton shower. Instead of aiming to calculate distributions analytically, we will derive approximate probability distributions for additional emissions of partons, and use them to formulate a Markov chain. The phase space regions formally connected to logarithmic divergences will now be connected

[6] see A.1 for an introduction to event shapes

to regions with a potentially unlimited number of emissions. The limitations of perturbative QCD lead to an infrared cut-off, thereby constraining the number of possible emissions. The parton shower therefore generates events, not analytical distributions. A distribution of any observable would have to be obtained by histogramming the resulting events.

Most of this chapter is based on [52, chapter 2.8]. As a starting case, in section 2.1 we introduce a QED (i.e. Abelian) example to demonstrate the basic features of a parton shower. There we will show how an approximation to the cross section after adding an additional photon emission to a process can be related to the original cross section. The corresponding generalizations needed to obtain cross sections for processes with an arbitrary number of additional photons are given in section 2.2. The modifications when describing a non-abelian theory, QCD, are given in section 2.3. Section 2.4 shows that for certain types of emissions, the formulation as a Markov chain is inappropriate and describes the special treatment for this type of emissions. The generalization to include emission in the initial state are given in section 2.5. The connection of parton showers and the evolution of parton distribution functions is briefly outlined in section 2.6. After that, we discuss the implementation of parton showers as Markov chains in section 2.7. The theory of matching matrix elements and parton showers is closely related to the theory of parton showers and therefore also relevant for this work, we introduce the basic concepts of matching in section 2.8. After that, we describe the three most widely used matching schemes in section 2.9.

## 2.1 Comparing Matrix Elements

We assume that a matrix element for a process $i \to ef$, where $e$ is an electron and $i$ and $f$ denote the particles in the initial state and the remaining particles in the final state, respectively, can be written in the form

$$\mathcal{M}^{i\to ef} = \bar{u}(p,\kappa)\mathcal{A}^{i\to ef}(p,p_j)$$

with $\bar{u}$ as the spinor of the electron and $\mathcal{A}$ denoting the remainder of the matrix element, being a function of the electron's momentum $p$ and the set of other particles' momenta $p_j$.

The matrix element for the process in which the aforementioned electron in the final state emits an additional photon with momentum $k$ is then given by adding the intermediate electron propagator of momentum $p' = p + k$, the additional electron-photon vertex and the polarization vector for the additional photon:

$$\mathcal{M}^{i\to e\gamma f} = -e\bar{u}(p,\kappa')\not{\epsilon}(k,\lambda)\frac{\not{p}+\not{k}+m}{(p+k)^2-m^2+i\epsilon}\mathcal{A}^{i\to ef}(p+k,p_j).$$

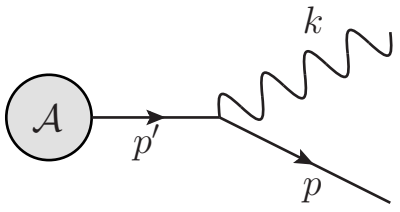


Figure 2.1: Photon emission off an electron in the final state.

In the parton shower framework the momenta of the final electron and photon and of the intermediate electron propagator are parameterized as shown below. This parameterization depends on two values, a value $x$ that describes the distribution of energy among the final particles and the transverse momentum $\mathbf{k}_\perp = (k^1, k^2, 0)$ of the final particles with respect to the intermediate electron. $\mathbf{k}_\perp^2$ measures the *scale* of the branching, a high $\mathbf{k}_\perp^2$ represents a hard branching while a low $\mathbf{k}_\perp^2$ represents a soft-collinear branching. As we want to probe the collinear divergence, we will assume $\mathbf{k}_\perp^2$ to be small. Using this parameterization, the momenta are given by

$$p'^\mu = p^\mu + k^\mu = \left(E + \frac{\mathbf{k}_\perp^2}{2x(1-x)E}, 0, 0, E\right), \tag{2.1}$$

$$p^\mu = \left(xE + \frac{\mathbf{k}_\perp^2}{4x(1-x)E}, 0, 0, xE - \frac{\mathbf{k}_\perp^2(1-2x)}{4x(1-x)E}\right) - k_\perp^\mu, \tag{2.2}$$

$$k^\mu = \left((1-x)E + \frac{\mathbf{k}_\perp^2}{4x(1-x)E}, 0, 0, (1-x)E + \frac{\mathbf{k}_\perp^2(1-2x)}{4x(1-x)E}\right) + k_\perp^\mu. \tag{2.3}$$

With these definitions the intermediate electron is set off-shell

$$p'^2 = \frac{\mathbf{k}_\perp^2}{x(1-x)} + \mathcal{O}\left(k_\perp^4\right), \tag{2.4}$$

while the final electron and photon remain massless[7],

$$k^2 = 0 + \mathcal{O}\left(k_\perp^4\right), \qquad p^2 = 0 + \mathcal{O}\left(k_\perp^4\right).$$

Up to terms of the order $\mathcal{O}\left(k_\perp^2\right)$ the numerator of the propagator can be replaced by the polarization sum of on-shell fermions with momentum $p' = p + k$,

$$\not{p}' = \sum_{\kappa'} u(p', \kappa')\bar{u}(p', \kappa').$$

[7]For so far, all final particles are assumed to be massless.

This results in

$$\begin{aligned}\mathcal{M}^{i\to e\gamma f} &= -e\sum_{\kappa'}\bar{u}(p,\kappa)\not{\epsilon}^*(k,\lambda)u(p',\kappa')\bar{u}(p',\kappa')\frac{x(1-x)}{\mathbf{k}_\perp^2}\mathcal{A}^{i\to ef}(p,p_j)\\ &\quad +\mathcal{O}\left(\mathbf{k}_\perp^2\right).\end{aligned} \tag{2.5}$$

Using the explicit expressions for the circular polarization vectors

$$\epsilon_\pm^{*\mu}(k)=\frac{1}{\sqrt{2}}\left(\frac{|\mathbf{k}_\perp|\,e^{\mp i\phi}}{2(1-x)E},1,\mp i,-\frac{|\mathbf{k}_\perp|\,e^{\mp i\phi}}{2(1-x)E}\right)$$

and the helicity Dirac spinors in the Weyl representation

$$\begin{aligned}u(p',-)&=\left(0,0,\frac{|\mathbf{k}_\perp|e^{-i\phi}}{\sqrt{2xE}},\sqrt{2xE}\right)^T,\\ u(p',+)&=\left(\sqrt{2xE},\frac{-|\mathbf{k}_\perp|e^{i\phi}}{\sqrt{2xE}},0,0\right)^T,\\ u(p,-)&=\left(0,0,0,\sqrt{2xE}\right)^T,\\ u(p,+)&=\left(\sqrt{2xE},0,0,0\right)^T,\end{aligned}$$

the summation over the photon polarizations yields

$$\sum_{\lambda=\pm}|\mathcal{M}^{i\to e\gamma f}|^2=2e^2\frac{1-x}{x\mathbf{k}_\perp^2}\frac{1+x^2}{1-x}|\mathcal{M}^{i\to ef}|^2+\mathcal{O}(|\mathbf{k}_\perp|^{-1}).$$

Writing this as a cross section and introducing the integral over the additional one-particle phase space yields

$$\begin{aligned}\mathrm{d}\sigma^{i\to e\gamma f}(p,p_j) &= 2e^2\int\frac{\mathrm{d}^3k}{(2\pi)^3 2E_k}\frac{x(1-x)}{\mathbf{k}_\perp^2}\frac{1+x^2}{1-x}\times\\ &\qquad\left[\frac{1}{x}\mathrm{d}\sigma^{i\to ef}(p',p_j)+\mathcal{O}(|\mathbf{k}_\perp|)\right].\end{aligned} \tag{2.6}$$

The differential over the photon phase space can be written as

$$\frac{\mathrm{d}^3k}{(2\pi)^3 2E_k}=\frac{\mathrm{d}x\;\mathrm{d}\phi\;\mathrm{d}\mathbf{k}_\perp^2}{4(2\pi)^3(1-x)}+\mathcal{O}(\mathbf{k}_\perp^2).$$

The differential cross section is therefore

$$\mathrm{d}\sigma^{i\to e\gamma f}(p,p_j)=\frac{\alpha}{2\pi}\int\frac{\mathrm{d}\mathbf{k}_\perp^2}{\mathbf{k}_\perp^2}\int\mathrm{d}x\left[\frac{1+x^2}{1-x}\mathrm{d}\sigma^{i\to ef}(p',p_j)+\mathcal{O}(|\mathbf{k}_\perp|)\right]. \tag{2.7}$$

In this equation the soft and collinear logarithmic singularity can be seen in the factor $\int \frac{\mathrm{d}\mathbf{k}_\perp^2}{\mathbf{k}_\perp^2}$: assuming a massless electron the lower limit in $|\mathbf{k}_\perp|$ would be zero, so the integration over $|\mathbf{k}_\perp|$ would be logarithmically divergent. The integration over $x$ is trivially limited by the constraint $0 \leq x \leq 1$. Moreover, the integration has to be constrained even further, as the limit $x \to 0, 1$ corresponds to soft singularities. Thus, every parton shower includes a cut on the $x$-variable, so that emissions outside the allowed region are classified as unresolvable. The actual cut is a choice that has to be made when implementing the parton shower, different choices are possible.

The differential cross section can also be written in the form

$$\mathrm{d}\sigma^{i\to e\gamma f}(p, p_j) = \frac{\alpha}{2\pi} \int_{\mathbf{k}^2_{\perp\,\min}}^{\mathbf{k}^2_{\perp\,\max}} \frac{\mathrm{d}\mathbf{k}_\perp^2}{\mathbf{k}_\perp^2} \int \mathrm{d}x \left[ P_{e\to e\gamma}(x) \mathrm{d}\sigma^{i\to ef}(p', p_j) + \mathcal{O}(|\mathbf{k}_\perp|) \right], \tag{2.8}$$

introducing the *splitting function* $P_{e\to e\gamma}(x)$ for the emission of a photon by an electron:

$$\begin{aligned}\mathrm{d}\sigma^{i\to e\gamma f}(p, p_j) = \frac{\alpha}{2\pi} \log\left(\frac{\mathbf{k}^2_{\perp\,\max}}{\mathbf{k}^2_{\perp\,\min}}\right) \int \mathrm{d}x P_{e\to e\gamma}(x) \mathrm{d}\sigma^{i\to ef}(p', p_j) \\ + \text{non-logarithmic terms.}\end{aligned}$$

The upper limit $\mathbf{k}_{\perp\,\max}$ is determined by phase space boundaries. For calculations regarding massless electrons the lower limit $\mathbf{k}_{\perp\,\min}$ is zero, reflecting the collinear singularity. When taking the non-vanishing electron-mass into account, this regularizes the singularity and the effective lower limit is of the order of the electron mass. Then the differential cross section is of the form

$$\begin{aligned}\mathrm{d}\sigma^{i\to e\gamma f}(p, p_j) = \frac{\alpha}{2\pi} \log\left(\frac{\mathbf{k}^2_{\perp\,\max}}{m_e^2}\right) \int \mathrm{d}x P_{e\to e\gamma}(x) \mathrm{d}\sigma^{i\to ef}(p', p_j) \\ + \text{non-logarithmic terms.}\end{aligned}$$

The term $\log\left(\frac{\mathbf{k}^2_{\perp\,\max}}{m_e^2}\right)$ is a remnant of the singularity and becomes important for energies much higher than $m_e$. The other singularity in this equation, namely for $x \to 1$ in $P_{e\to e\gamma}(x) = \frac{1+x^2}{1-x}$ is a left-over of the soft photon singularity that is compensated for by virtual photon corrections [52, chapter 2.8.3.2] and usually cut by restraining the allowed $x$-interval.

In this derivation, we suppressed the dependence on the azimuthal angle $\phi$. Thus the emissions are distributed uniformly over the azimuthal angles, which would correspond only to an additional factor of $\int \frac{\mathrm{d}\phi}{2\pi}$. Including angular correlations is one way of improving the parton shower, but is not yet pursued in our approach.

The parameterization in equations (2.1) - (2.3), using $\mathbf{k}_\perp$ as the scale of the emissions, is not the only possible parameterization. Due to the form of (2.7), variables that conserve the logarithmic differential $\frac{\mathrm{d}\mathbf{k}_\perp^2}{\mathbf{k}_\perp^2}$ can be used as a scale variable. Changing the variable used as scale will only lead to differences in higher orders. One popular choice for the scale variable and the choice we will use in our implementation is the virtuality $t$ of the intermediate parton. Virtuality is defined as the squared four-momentum minus the particle's rest mass. It is thus a measure for the off-shellness and hence describes a particle's ability to undergo additional branchings. From (2.4) it follows that $t = p'^2$ and $\frac{\mathrm{d}t}{t} = \frac{\mathrm{d}\mathbf{k}_\perp^2}{\mathbf{k}_\perp^2}$.

## 2.2 Multiple Emissions

The results of the previous section can be generalized to the emission of multiple photons as well, that is to the process $\mathrm{d}\sigma^{i\to en\gamma f}$ with an arbitrary number $n$ of photons. The emitted photons are assumed to be ordered in transverse momentum, $\mathbf{k}_\perp^2$. This is based on the assumption that hard emissions happen on shorter timescales and thus cannot be influenced by the soft emissions happening on longer timescales. The probability for the emission of a photon with a given transverse momentum can be read from equation (2.8), it is at leading order in $|k_\perp|$

$$\frac{\alpha}{2\pi\mathbf{k}_\perp^2}\int \mathrm{d}x\left[P_{e\to e\gamma}(x)\mathrm{d}\sigma^{i\to ef}(p',p_j)\right]. \tag{2.9}$$

The corresponding conditional probability $P_\gamma(\mathbf{k}_\perp^2)$ needs to be normalized to the cross section of the hard process, therefore it is

$$P_\gamma(\mathbf{k}_\perp^2) = \frac{\mathrm{d}\sigma^{i\to e\gamma f}(p',p_j)\big|_{\mathbf{k}_\perp^2}}{\mathrm{d}\sigma^{i\to ef}(p,p_j)} \simeq \frac{\alpha}{2\pi\mathbf{k}_\perp^2}\int \mathrm{d}x P_{e\to e\gamma}(x), \tag{2.10}$$

assuming that the differential cross section is continuous for small momenta of the emitted photon, $\mathrm{d}\sigma^{i\to e\gamma f}(p',p_j) \simeq \mathrm{d}\sigma^{i\to e\gamma f}(p,p_j)$.

But this conditional probability is not correct, as an emission at a greater scale could have happened before and therefore the scales of the emissions would not be ordered. The relevant probability is the probability for the emission at $\mathbf{k}_\perp^2$ being the hardest emission. Therefore the probability in equation (2.10) has to be weighted with the probability $\Delta(\mathbf{k}_\perp^2, \mathbf{k}_\perp^{2\,max})$ that no emission at a scale greater than $\mathbf{k}_\perp^2$ occurred. Obviously, the probability satisfies the starting condition

$$\Delta(\mathbf{k}_\perp^{2\,max}, \mathbf{k}_\perp^{2\,max}) = 1. \tag{2.11}$$

For a differential change in $\mathbf{k}_\perp^2$, the probability for an emission is given by (2.10). The differential equation for $\Delta$ is then

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{k}_\perp^2}\Delta(\mathbf{k}_\perp^2, \mathbf{k}_\perp^{2\,max}) = -\frac{\alpha}{2\pi\mathbf{k}_\perp^2}\int \mathrm{d}x P_{e\to e\gamma}(x). \tag{2.12}$$

The solution to this differential equation is

$$\Delta(\mathbf{k}_\perp^{2\,min}, \mathbf{k}_\perp^{2\,max}) = \exp\left[-\int_{\mathbf{k}_\perp^{2\,min}}^{\mathbf{k}_\perp^{2\,max}} \frac{\mathrm{d}\mathbf{k}_\perp^2}{\mathbf{k}_\perp^2}\int \mathrm{d}x \frac{\alpha}{2\pi} P_{e\to e\gamma}(x)\right] \tag{2.13}$$

and gives the probability for an electron not to emit a photon with a transverse momentum between $\mathbf{k}_\perp^{2\,min}$ and $\mathbf{k}_\perp^{2\,max}$.

This no-emission probability is the central variable of a parton shower, it is called the *Sudakov form factor* [58]. The complete probability for a branching at hardness $\mathbf{k}_\perp^2$ to be the hardest is therefore

$$P_\gamma^{hardest}(\mathbf{k}_\perp^2) = \frac{\alpha}{2\pi\mathbf{k}_\perp^2}\left[\int \mathrm{d}x P_{e\to e\gamma}(x)\right]\cdot\Delta(\mathbf{k}_\perp^2, \mathbf{k}_\perp^{2\,max}). \tag{2.14}$$

This equation can then be used to iteratively calculate the respective next hardest branching. Implementing (2.14) in a Markov chain Monte Carlo simulation would be the simplest implementation of a QED parton shower.

## 2.3 Generalization to QCD

In section 2.1 we calculated the splitting function for the emission of a photon from an electron

$$P_{e\to e\gamma}(x) = \frac{1+x^2}{1-x} \tag{2.15}$$

as a side-product. The splitting function for an emission off a quark is proportional to $P_{e\to e\gamma}(x)$ with an additional factor accounting for the fractional electric charge,

$$P_{q\to q\gamma}(x) = |Q_q|^2\,\frac{1+x^2}{1-x}. \tag{2.16}$$

A similar calculation using QCD can be performed and leads to the splitting functions for QCD that read (see e.g. [53, chapter 5.1]):

$$P_{q\to qg}(x) = C_F \frac{1+x^2}{1-x}, \tag{2.17}$$

$$P_{\bar{q}\to\bar{q}g}(x) = P_{q\to qg}(x), \tag{2.18}$$

$$P_{g\to gg}(x) = C_A \left[\frac{1-x}{x} + \frac{x}{1-x} + x(1-x)\right], \tag{2.19}$$

$$P_{g\to q\bar{q}}(x) = T_R \left[x^2 + (1-x)^2\right] \quad \text{for every kinematically allowed quark flavor,} \tag{2.20}$$

with the $SU(3)$-factors $C_A = N_C = 3$, $C_F = \frac{N_C^2-1}{N_C} = \frac{4}{3}$ and $T_R = \frac{1}{2}$.

The Sudakov factor needs to be adjusted to the increased number of possible branchings. The splitting functions for all possible branchings, e.g. for a gluon these are the branchings into two gluons or all kinematically allowed quark pairs, add up. Thus the splitting function for a gluon to branch is given by the splitting function

$$P_{g\to anything}(x) = P_{g\to gg}(x) + \sum_{\substack{q\in\text{allowed}\\ \text{flavors}}} P_{g\to q\bar{q}}(x) \tag{2.21}$$

and the corresponding Sudakov factor for a gluon reads

$$\Delta(t^{min}, t^{max}) = \exp\left[-\int_{t^{min}}^{t^{max}} \mathrm{d}t \int \mathrm{d}x \frac{\alpha_S}{2\pi t}\left(P_{g\to gg}(x) + \sum_{\substack{q\in\text{allowed}\\ \text{flavors}}} P_{g\to q\bar{q}}(x)\right)\right]. \tag{2.22}$$

Two remarks concerning the Sudakov factors like the one given in (2.22) are in order. So far, we only denoted the coupling constant by $\alpha_S$, omitting any scale dependence. A straight-forward choice for the scale used in the the calculation of $\alpha_S$ would be the evolution scale $t$ of the parton shower. A more sophisticated calculation [59] favors the usage of a running scale[8] $x(1-x)t$. This approach was found to give a better representation of higher order contributions. Moreover, we did not specify borders for the integral over $x$. For the definition we will use in our implementation the borders will correspond to soft singularities, we will therefore have to choose a cut-off for the integral (cf. section B.2.1).

[8] This running scale is equal to the squared transverse momentum of the branching's outgoing partons if $x$ is interpreted as the ratio of the light-cone momentum $p_+ = E + p_Z$ (with the $z$-axis along the direction of flight of the incoming parton).

## 2.4 Color Coherence and Angular Ordering

So far, the parton shower was formulated as a Markov chain of independent emissions and interference terms were disregarded. This has been expected to be a good approximation because collinear emissions will mostly only be collinear to only one external leg, thus all interference terms will be less divergent and only contribute at next-to-leading orders. However, there is one case, where interference terms become important. This is the case when the parton shower generates an emission under a relatively large angle. Then the argument, that all interference contributions are suppressed by their lower collinearity no longer holds and the assumption that each particle radiates independently can no longer be applied. For example, consider a photon that splits into a quark-antiquark pair where one of the quarks subsequently emits a gluon. If the gluon is emitted under a relatively large angle, it will not be able to resolve the individual color charges of the quark pair, but see the combined charge instead. The combination of the two quarks leads to a color singlet, which means that emissions of this kind cannot occur. In this case, the interference is destructive and as large as the individual contributions. A more sophisticated calculation [60] shows that for a parton shower this *color coherence* can be implemented by vetoing emissions with an opening angle larger than the opening angle of the previous branching. Observables that are particularly sensitive to angular ordering are the kinematical correlations between jets in multi-jet events. For example, first evidence for color coherence was found in the kinematical correlation between the second and third hardest jet in $p\bar{p}$-collisions [61].

A similar effect has been known for a long time in cosmic ray physics as the decrease of ionization by narrow electron-positron pairs, named *Chudakov effect* [62].

## 2.5 Initial-State Radiation and Backwards Evolution

Up to now, we treated all evolution to be forward in causality, which means that for every parton the partons it emits are generated. Applied to a collision at a hadron collider this would lead to an algorithm where the evolution starts with two constituent partons from either hadron, simulate these partons' shower and use two residual partons, one from each side, as input to the hard scattering. This method has the drawback, that as no information about the hard scattering is used, mostly the partons at the end of the shower will be unsuitable for the hard scattering for example by either having to few momentum or by having a wrong flavor. Therefore a lot of tries would be needed to produce a fitting initial state, wasting enormous amounts of

computing time.

The solution to this problem is the *backwards evolution* [63]. In contrast to the forward evolution, here the parton that emitted the given parton is generated. The evolution starts with the partons in the initial state of the hard scattering and evolves them back to the constituent partons in the hadron. Partons in the initial state of the hard interaction are not chosen randomly, but according to the parton distributions, and with the cross section:

$$\sigma_X = \sum_{i,j} \int \mathrm{d}x_1 \mathrm{d}x_2 f_i\left(x_1, Q^2\right) f_j\left(x_2, Q^2\right) \hat{\sigma}_{ij\to X}\left(p_1, p_2, Q^2\right), \tag{2.23}$$

where $\hat{\sigma}_{ij\to X}$ is the partonic cross section. Note that this description does not violate causality. The parton distribution functions $f\left(x, Q^2\right)$ encapsulate all forward evolutions and the backwards evolution merely chooses one of these evolutions.

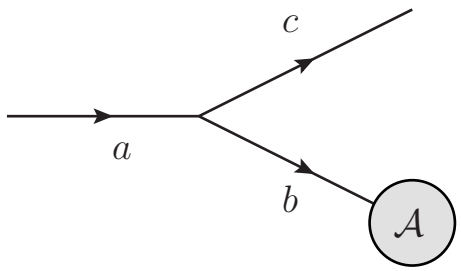


Figure 2.2: Emission in the initial state: For the parton $b$, its mother parton $a$ and the emitted parton $c$ are generated. The gray circle $\mathcal{A}$ stands for the remaining event, possibly including already found emissions in the initial state.

Consequently, during the simulation of the initial-state shower, the parton distribution functions have to be taken into account when simulating branchings in the backwards evolution. The probability for a branching is by definition proportional to the parton density of the branching parton. Thus every branching is weighted with the branching parton's PDF. For normalization, this is accompanied by a division by the parton distribution function of the newly created parton. The Sudakov factor for a parton of type $b$ takes

the form

$$\Delta_b^{ISR}(t_a, t_b) = \exp\left[ -\int_{|t_a|}^{|t_b|} \mathrm{d}t' \int_0^1 \frac{\mathrm{d}x}{x} \frac{\alpha_S}{2\pi t'} \sum_{a,c} \left( P_{a\to bc}(x) + P_{a\to cb}(x) \right) \frac{f_a\left(\frac{x_b}{x}, t'\right)}{f_b(x_b, t')} \right] \quad (2.24)$$

with an analogous generalization to multiple branchings. This prescription ensures that the probabilities to find a final mother parton will be distributed according to the parton distribution function for a parton of the same type, especially it is prohibited to find partons with a momentum fraction $x > 1$, which could occur when using a naive Sudakov factor.

For the Sudakov factors for ISR, remarks similar to those for the FSR Sudakov factor (2.22) are in order: For the argument of the running coupling, the running scale $(1 - x)t$ was found [60] to give the best representation of higher-order contributions. For the borders of the integral over $x$ the situation is different to the one in FSR due to the inclusion of PDFs. The limit $x \to 0$, where the current parton would be created in a soft emission, cannot be the source of a divergence, since the PDF $f_a\left(\frac{x_b}{x}, t'\right)$ will be zero once $\frac{x_b}{x} > 1$. The other limit, $x \to 1$, which corresponds to the emission of a soft particle, is regularized by a cut-off that has to be chosen manually, usually it is demanded that the emitted parton has a certain minimum energy.

The generation of an additional branching according to (2.24) also includes the generation of the other particle produced in the branching, here denoted as $c$. This parton is added to the final state and can initiate a final-state parton shower on its own.

## 2.6 Comparison to DGLAP

The evolution of parton distribution functions $q(x, t)$ with varying scale $t$ is described by the Dokshitzer-Gribov-Lipatov-Altarelli-Parisi (DGLAP) evolution equations [64, 65, 66]. They can be written in the form

$$t\frac{\partial}{\partial t} q(x, t) = \frac{\alpha_S(t)}{2\pi} \int_x^1 \frac{\mathrm{d}\xi}{\xi} P\left(\frac{x}{\xi}\right) q(\xi, t). \quad (2.25)$$

The previous section has shown the close connection between an initial-state parton shower at a hadron collider and the evolution of parton distribution functions. The connection between parton showers and DGLAP evolution can be stated as

> A parton shower is an exclusive solution of the DGLAP-evolution equations starting with single partons.

So for example, the final-state radiation in an $e^+e^- \to u\bar{u}$ interaction can be seen as a DGLAP-evolution starting from a parton distribution functions

$$u(x, t_{hard}) = \bar{u}(x, t_{hard}) = \delta(x - 1)$$

for the two imaginary hadrons produced in the interaction, each consisting only of the $u$ or $\bar{u}$ quark to the lower cut-off scale $t_{cut}$.

Nevertheless, there are some differences. While solutions to the DGLAP equations are inclusive solutions, parton showers are constructed to give exclusive solutions. Moreover the parton distribution functions are evolved in the infinite momentum frame treating all particles collinear, while partons acquire transverse momentum during the evolution in the parton shower.

## 2.7 Implementation in a Markov Chain Monte Carlo

The implementation of the parton shower in a Markov chain Monte Carlo is as follows: For a given particle at a scale $t_{max}$ the scale of its next branching is found by solving the equation

$$R = \Delta(t, t^{max}), \tag{2.26}$$

where $R$ is a random number distributed uniformly in the interval $[0 : 1]$ and $\Delta$ is the Sudakov factor for either FSR or ISR. Solving this equation for $t$ gives the scale of the next branching. The corresponding energy ratio can be found by solving the equation

$$R \int_{z_-}^{z_+} \mathrm{d}z' \, \frac{\alpha_S}{2\pi} P(z') = \int_{z_-}^{z} \mathrm{d}z' \, \frac{\alpha_S}{2\pi} P(z') \tag{2.27}$$

for $z$. $z_-$ and $z_+$ are the limits chosen for the $z$ integral. In the case of multiple possible types of branchings, one of these branchings is chosen according to a weight given by either the respective splitting function for FSR or the product of splitting function and the branching parton's PDF, $P_{a\to bc}(z) f_a(x_a, t_a)$, for ISR. The particles' momenta can then be inferred from their respective virtualities, the energy ratio $z$ and the randomly chosen azimuthal angle $\phi$.

The evolution starts at a high scale and evolves down to a cut-off scale. The cut-off scale has to be chosen as a border to the hadronization. It is usually chosen to be of the order of about $1\,\mathrm{GeV}^2$. The starting scale is given by the center-of-mass energy for the final-state radiation in $s$-channel

processes. For other processes different choices can be made. Thus the evolution for the FSR is from $t_{start} = s$ down to $t_{cut} \sim 1\,\mathrm{GeV}^2$. For the initial-state radiation, the evolution is from a low negative virtuality $t_{start} = -s$ up to the negative of the cut-off scale $t_{cut}^{ISR} = -t_{cut}$.

## 2.8 Combining Matrix Elements and Parton Showers

The description of multiple emissions via the matrix element and via the parton shower are two complementary ways of describing the event.

Matrix elements contain the complete information up to a given order in $\alpha_S$. Their disadvantage is the rapid growth of complexity with the order of $\alpha_S$ and the number of particles. The complexity of a parton shower on the other hand grows only linearly with the number of particles but parton showers are only correct in the soft/collinear limit. It is not feasible to describe partonic events by using only matrix elements and inadequate to try to use only the simplest matrix element with a subsequent parton shower. The first approach will suffer from the rise of the needed computing time, so that the computation of a reaction at any collider, where the number of partons produced will typically be of the order $\mathcal{O}(100)$, is out of reach. The pure parton shower description will suffer from the poor description of physics away from the soft and collinear limit.

Thus, it is desirable to combine these two approaches, using parton showers for *jet substructure* and matrix elements for *hard, widely-separated jets*[9]. Combining these approaches is a non-trivial task. There is no immanent distinction whether an additional emission belongs to the regime of hard, widely separated jets or merely belongs to the jet-substructure regime.

The need to build a more sophisticated procedure to combine these two approaches has led to the introduction of *matching procedures*[10]. A matching procedure intends to combine the description of up to a certain number of multiple, widely separated jets by the matrix element and the description of possible additional jets and the substructure of the jets by a parton shower.

[9] Strictly speaking, this leads to a *slicing-based matching procedure*. We restrain ourselves to these approaches here. See e.g. [51] for on overview of different matching approaches, namely slicing-based, *unitarity-based* and *subtraction-based*.

[10] In the past it was not differentiated between the terms "matching" and "merging". Some authors [67] proposed to use "merging" for procedures where information from the parton shower is incorporated in the matrix element or vice versa while using "matching" for procedures that combine matrix elements and parton showers without changing them. Recently it has come up to use "merging" for combining leading-order matrix elements and parton showers while using "matching" for combining NLO matrix elements and parton showers. In this thesis we will only use the term "matching".

The main approaches for matching to leading order calculations are the CKKW [68, 69], CKKW-L [70] and MLM [71, 72] schemes, for a general overview see [67, 73, 74]. For matching to NLO matrix elements, some new approaches have recently been developed, the two most widely used are MC@NLO [75] and POWHEG [76, 77]. As `WHIZARD` at the present stage only uses leading-order matrix elements, we restrain ourselves to the leading-order matching procedures.[11]

## 2.9 Overview of Matching Procedures

In this section we will give an overview of the matching schemes introduced in the previous section. The following steps are common to all leading-order matching procedures. We take a process of the form

$$(initial\ state) \to (final\ state)$$

as our generic example.

1. A maximal number $N$ of additional jets to be simulated by the matrix element is chosen.

2. For every $n$ with $n = 0, 1, \ldots N$, the matrix elements for the processes

   $$(equivalent\ initial\ state) \to (final\ state) + n\ jets$$

   are generated and the cross sections $\sigma_n$ with respect to certain given cuts are calculated. Equivalent initial states are initial states of processes that can be obtained by adding up to $n$ parton branchings to the original process, e.g. by adding a $g \to q\bar{q}$ branching to the process $q\bar{q} \to e^+e^-$ the process $qg \to e^+e^-q$ is generated, thus for $n = 1$ $qg$ is an equivalent initial state for the process $q\bar{q} \to e^+e^-$.

3. Events described solely by the matrix elements are generated. The probability to generate an event with $n$ additional jets is given by

   $$P(n) = \frac{\sigma_n}{\sum_m \sigma_m}. \tag{2.28}$$

4. These generated events can then be accepted/rejected/reweighted according to the kinematical configuration of the event either with or without a parton shower applied.

[11] The inclusion of NLO matrix elements is currently under development. First steps to a dipole subtraction scheme along the lines of the Catani-Seymour subtraction scheme [78, 79] are implemented. Moreover an interface to the GoSam [80] package is in preparation.

The matching schemes differ in the following ways

1. The cuts applied to the single exclusive matrix elements can be different.

2. The procedure to accept/reject/reweight the events can be applied before or after the parton shower.

3. The procedure to accept/reject/reweight the events can be different.

4. The settings for the parton shower can be different. The parton shower can be modified to veto specific types of branchings.

We will describe the three main schemes for leading-order matching in the following. The differences between these approaches have been investigated for $W$-production at the Tevatron and the LHC [73, 74]. Despite their algorithmic differences, a general consistency with occasional discrepancies has been found. However, it is not clear, if these results also apply to arbitrary processes.

### 2.9.1 MLM

The MLM [71, 72] matching procedure is a phenomenological ansatz and the simplest of the common matching procedures. Its main aim is to describe jets according to the matrix element calculation while describing the substructure of these jets using the parton shower. To achieve this, events containing significant interplay between these domains are vetoed. A matching procedure implementing the MLM matching will use matrix-element-level events and perform an unmodified parton shower. After the parton shower it is checked that the jet topology of the showered and unshowered event "match", that means that

- the number of jets after the shower is equal to the number of the hard partons given by the matrix element. This means that neither a parton splits into partons that were not hard enough to be recognized as a jet, nor that one parton emitted a parton hard enough to be the seed for a jet on its own. The events where such branchings occur would then be intended to be described by the $(n-1)$- or $(n+1)$-jet matrix element, thus these events are vetoed. There is one exception to this rule: for the highest jet multiplicity, $(n = N)$, additional jets are allowed to be described by the parton shower, as the corresponding matrix elements were not taken into account in the first place.

- for all jets after the matrix element, a "matched" jet after the shower is found. The definition when jets "match" is dependent on the implementation, for example jets can be regarded as matched when their angular separation is below some threshold. The implementation in WHIZARD will use a definition on jet scales, to be explained later.

If this is the case, the event is accepted, otherwise it will be discarded and the simulation will proceed with a new matrix element event.

The MLM matching scheme is the simplest way to implement a matching procedure, as the three steps, matrix element, parton shower and matching, are completely disjoint, therefore the matching requires no modification of the matrix element calculation or the parton shower.

We implemented one version of the MLM matching procedure in WHIZARD. This implementation will be described in greater detail in section 4.5.

### 2.9.2 CKKW

The CKKW approach was originally presented by Catani, Krauss, Kuhn and Webber [68] for $e^+e^-$-collisions and later extended to hadronic collisions by Krauss [69]. The central idea of CKKW is to use the full matrix element for all emissions above some jet scale $y_{ini}$[12] and the parton shower for all emissions below that scale. The steps are as follows:

1. Matrix-element events are generated according to the cross sections $\sigma_n$ using a fixed $\alpha_S = \alpha_S^{ME}$.

2. Using the $k_T$ clustering algorithm a pseudo-parton shower history is generated. This means that starting from the $2 \to n$ process, iteratively the two partons that are most likely to have been produced in a parton shower branching are replaced by their supposed mother particle[13]. This procedure stops when a $2 \to 2$ process is reached or when no parton shower branchings are possible anymore (e.g. a $qg \to q'W \to 3\,jets$ event.). Thus the scales $q_i$ for each branching can be inferred from the jet scales at which the branching occur.

3. A coupling constant weight of $\prod_i \alpha_S(y_i) / \left(\alpha_S^{ME}\right)^{n-2}$ is calculated, correcting for the fixed coupling constant in the matrix element.

[12] See section A.2 for an introduction to jet scales and the $k_T$ clustering algorithm.

[13] In addition, some implementations also allow quark-antiquark pairs to be replaced by a $Z$ or $W$ boson.

4. For every internal and external line, a weight given by a suitable combination of Sudakov factors is calculated. The task of this weight is to take the probability to produce a hard emission in the parton shower into account.

5. The event is reweighted with the combined weight of the two previous steps.

6. The parton shower is invoked. The starting scale of the shower is set according to the jet scale of the branching in the pseudo-parton shower history, where the particle was produced. Moreover, any branching that would give an emission above $y_{ini}$ is to be vetoed.

### 2.9.3 CKKW-L

The CKKW-L algorithm is a variant of the CKKW scheme that was proposed by Lönnblad [70]. The differences compared to the CKKW approach are in the generation of the pseudo-parton shower history and in the application of Sudakov weights. While in the CKKW approach only the most likely history is generated, in the CKKW-L all possible shower histories are generated, out of which one history is selected randomly with probabilities proportional to the product of branching probabilities. The Sudakov weights are not calculated analytically, but by using trial emissions: for every intermediate $2 \to n', n' < n$ process, a parton shower is invoked, the starting scale of the shower is set to the scale of the intermediate process. If the first emission in the shower leads to a $2 \to n' + 1$ event with a jet scale larger than the jet scale of the $2 \to n' + 1$ process in the shower history, the event is discarded.

### 2.9.4 Comparison of Matching Schemes

From the point of view of a theoretical particle physicist, the CKKW(L) approach should give a more reliable combination of matrix elements and parton showers. For the CKKW approach, the original publication claims that the dependence on the merging scale $y_{cut}$ cancels at next-to-leading logarithmic (NLL) accuracy. This proof makes use of a formalism, whose NLL accuracy was to be proven, but this proof was never published (cf. footnote 3 in [67]). Therefore it is doubtful if this approach can be considered to be independent of the merging scale at NLL. For the CKKW-L approach, it was proven that the dependence on $y_{ini}$ cancels for suitably defined parton showers. Moreover, the CKKW-L approach has the advantage that all corrections in the parton shower are automatically included in the matching. In contrast, the MLM approach is a purely phenomenological ansatz. Its main

advantage is its simplicity. The price for this simplicity is the need to veto events, making the generation of events less efficient.

However, comparison studies [67, 73, 74] have found good agreement between predictions based on CKKW, CKKW-L and MLM matching. It remains in question if this agreement can be expected for any combination of observable and processes.

# 3 The Analytic Parton Shower

As shown in section 2 the parton shower is a well-defined approximation of the full matrix element. Therefore, one important goal should be to be able to reconstruct the matrix element from the parton shower. In common parton shower algorithms, this ability is lost due to the formulation of the parton shower as a Markov chain in such a way, that branchings that fail to respect correct kinematics can be produced and are subsequently rejected or manually modified to respect momentum conservation. It is these branchings that prevent the probability for a branching to be calculated after the branching is generated. Therefore in developing an *analytic parton shower*, care was taken to avoid branchings that need to be rejected or manually modified, thereby preserving the ability to reconstruct the matrix element.

As already hinted at, the parton shower we are presenting here uses virtuality as its ordering variable, hence we will replace the variable $\mathbf{k}_\perp^2$ used in section 2 by the virtuality $t$. Moreover we will replace the value describing the distribution of energy $x$ by a variable called $z$ that is defined as the ratio between the energies of the first daughter and the mother parton, $z = \frac{E_b}{E_a}$. From equations (2.1) and (2.2) it follows that $z = x + \mathcal{O}\left(\frac{\mathbf{k}_\perp^2}{E_b E_c}\right)$.

Our work is based on an approach first presented in [47]. There, an analytic final-state parton shower was introduced. This parton shower was later used in the `GenEvA` framework [81, 48]. as a phase space generator. We review the presented algorithm in section 3.1 and present our extension in section 3.2. Section 3.3 describes an analytic parton shower for the initial state that was developed during this thesis. Finally, we discuss the possibility of a $p_T$-ordered analytic parton shower in section 3.4.

## 3.1 General Concept

The analyticity of a parton shower algorithm can be achieved by changing the way branchings are simulated, while maintaining the formulation of the parton shower as a Markov chain. The two main changes to the simulation are the simultaneous simulation of the branchings of sisters and replacing the splitting variable $z$, that normally is the ratio of the first daughter's energy or light-cone momentum to the mothers' energy or light-cone momentum, by an invariantly defined angle $\cos\theta$.

The first modification is to replace the simulation of individual branchings by the simulation of so-called double branchings. A double branching consists of the simultaneous branching (or no-branching) of the two daughter-partons of one parton. So instead of taking one parton $a$ and letting it branch into

two partons, $a \to bc$, an existing branching $a \to bc$ is replaced by the double branching $a \to bc \to defg$ with the new partons $d, e, f, g$ in case both partons $b$ and $c$ branch or the analogous branching combination for the cases where one or both of the daughters do not branch. The advantage is that the energy conservation[14]

$$\sqrt{t_a} \geq \sqrt{t_b} + \sqrt{t_c} \tag{3.1}$$

can be taken into account during the generation of the branchings, avoiding the production of complicated interconnections between different single branchings. The sequence of steps is then

- Pick a parton with unprocessed daughters $b$ and $c$.
- Generate $\{t_b, v_b\}$ and $\{t_c, v_c\}$ for branchings of both daughters independently, with the probability given by the single branching probability. Here, $v_i$ stands for the values needed to describe the branching of particle $i$ apart from the virtuality $t_i$, like the opening angle $\cos\vartheta_i$ (see below), the azimuthal angle $\phi_i$ and the type of the daughter parton.
- Keep the branching of the daughter with the higher scale $t_{max} = \max(t_b, t_c)$. Discard the branching of the other daughter.
- Determine new values for the other daughter with the maximum scale set to $t_* = \min\left[t_{max}, \left(\sqrt{t_a} - \sqrt{t_{max}}\right)^2\right]$. This ensures that the constraint given in equation (3.1) is automatically fulfilled.

For the different cases, the double branch probabilities can be constructed from the single branching probabilities $\mathcal{P}_i^{br}(t_i, v_i)$ given by the generalization of equation (2.14) and Sudakov factors $\Delta_i(t_a, t_i)$ for a branching at the scale $t_i$ and the remaining values $v_i$ and the probability $\mathcal{P}_i^{nb}$ for no branching above the cut-off. The double branch probabilities for the case in which both daughters branch is

$$\begin{aligned} \mathcal{P}^{br,br}(t_b, v_b, t_c, v_c) &= \theta(t_b - t_c)\mathcal{P}_b^{br}(t_b, v_b)\Delta_c(t_a, t_b) \\ &\quad \mathcal{P}_c^{br}(t_c, v_c; t_a = t_*) \\ &+ \theta(t_c - t_b)\mathcal{P}_c^{br}(t_c, v_c)\Delta_b(t_a, t_c) \\ &\quad \mathcal{P}_b^{br}(t_b, v_b; t_a = t_*), \end{aligned} \tag{3.2}$$

[14] The following equation can best be understood in the rest frame of the mother. Then $\sqrt{t_a}$ is the mother's mass and energy and trivially the sum of the daughters' masses $\sqrt{t_b} + \sqrt{t_c}$ has to be less.

while in the case that only one daughter branches

$$\mathcal{P}^{br,nb}(t_b, v_b) = \mathcal{P}_b^{br}(t_b, v_b)\Delta_c(t_a, t_b)\Delta_c(t_*, t_{cut}), \tag{3.3}$$
$$\mathcal{P}^{nb,br}(t_c, v_c) = \mathcal{P}_c^{br}(t_c, v_c)\Delta_b(t_a, t_c)\Delta_b(t_*, t_{cut}), \tag{3.4}$$

and in the case no parton branches

$$\mathcal{P}^{nb,nb} = \Delta_b(t_a, t_{cut})\Delta_c(t_a, t_{cut}). \tag{3.5}$$

Taking all different combinations into account, the full double branch probability can be composed in the following way

$$\begin{aligned}\mathcal{P}(t_b, v_b, t_c, v_c) &= \mathcal{P}^{br,br}(t_b, v_b, t_c, v_c) \\ &+ \mathcal{P}^{br,nb}(t_b, v_b) \quad \delta(t_c) \\ &+ \mathcal{P}^{nb,br}(t_c, v_c) \quad \delta(t_b) \\ &+ \mathcal{P}^{nb,nb} \qquad \delta(t_b)\delta(t_c),\end{aligned} \tag{3.6}$$

where $\delta(t_i)$ stands for a parton that is on-shell, thus not branching any further.

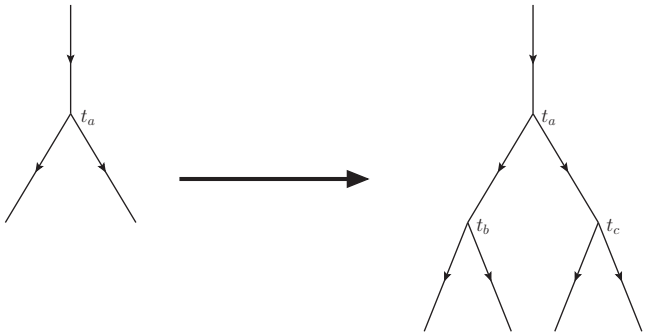


Figure 3.1: Schematic view of a double branching in FSR. Before the double branching (left): A parton has branched at a scale $t_a$ into two on-shell daughter partons. After the simulation of the double branching the branching scales $t_b, t_c$ for the daughter partons are known. In case the daughters branch themselves the values needed for their respective branchings are generated as well. The case in which both daughters branch, $t_b > t_{cut}, t_c > t_{cut}$, is shown on the right. For the next step the double branchings of the two daughter partons will be simulated, the branchings at $t_b$ and $t_c$, respectively, replace the parton branching at $t_a$ on the left.

The second modification is replacing the kinematic ratio $z$: Using an energy ratio is error-prone, potentially changing its value when applying Lorentz boosts. Hence, we replace it by the angle $\cos\theta$ in the mother's rest frame between the momentum of the first daughter and the boost axis. This leads to simple phase space limits

$$-1 \leq \cos\theta \leq 1.$$

There is a direct correspondence between the $\cos\theta$ angle and the energy splitting $z$ [47] as a function of the masses of the daughters $t_b$ and $t_c$:

$$z = \frac{1}{2}\left[1 + \frac{t_b}{t_a} - \frac{t_c}{t_a} + \beta_a \cos\theta_a \lambda(t_a, t_b, t_c)\right] \tag{3.7}$$

with the boost $\beta_a$ and the phase space factor $\lambda$:

$$\beta_a = \sqrt{1 - \frac{t_a}{E_a^2}} \qquad \text{and} \qquad \lambda(t_a, t_b, t_c) = \frac{1}{t_a}\sqrt{(t_a - t_b - t_c)^2 - 4t_b t_c}.$$

Schematically, this means that the integration over $z$ is replaced by an integration over $\cos\theta$ as follows:

$$\int \mathrm{d}z P(z) \Rightarrow \int \mathrm{d}\cos\theta \frac{\mathrm{d}z}{\mathrm{d}\cos\theta} P(z(\cos\theta)). \tag{3.8}$$

The Sudakov factor given in (2.22) thus transforms to

$$\Delta(t^{min}, t^{max}) = \exp\left[-\int_{t^{min}}^{t^{max}} \mathrm{d}t \int \mathrm{d}\cos\theta \frac{\mathrm{d}z}{\mathrm{d}\cos\theta} \frac{\alpha_S}{2\pi t} P_{g\to anything}(z(\cos\theta))\right]. \tag{3.9}$$

The important distinction between common and analytic parton showers is that in the analytic parton shower every branching is generated with a calculable probability. Every source for vetoing branchings where the probability for the veto cannot be calculated has therefore been avoided.

The simulation of the branching generates a value of $\cos\theta$ under the assumption that both daughters are massless $t_b = t_c = 0$. But a branching by one or both of the daughter partons will set the parton off-shell, $t_{b,c} > 0$, thus changing the value of $\cos\theta$ for the mother's branching. As the $\cos\theta$ value is considered to be the fundamental quantity, a procedure is implemented to restore the original value. This procedure is called *shuffling*. It redistributes the mothers' energy among the daughters and their daughters such that the original angles are restored. This can be achieved by redistributing the energy according to equation (3.7) using the updated values for $t_b$ and $t_c$.

## 3.2 Improved Analytic Final-State Parton Shower

The `GenEvA` framework [81, 48] is an event generation framework designed to combine matrix elements and parton showers during event generation. It uses its parton shower to distribute events over phase space, in order to reweight them to a corrected distribution later. Therefore only a simplified implementation of parton showers was included in the framework, as the reweighting would later reintroduce the correct distributions. We, on the other hand, will use the analytic parton shower to generate exclusive events and cannot defer anything to a reweighting procedure, we are therefore forced to implement the full theory of parton showers. The two main simplifications made in the `GenEvA` framework are the omission of the running of the coupling constant and the omittance of color coherence. Our extensions to the parton shower are as follows.

The running of the coupling constant $\alpha_S(Q^2)$ was implemented, the inclusion is straight-forward. The coupling was chosen to be

$$\alpha_S = \alpha_S\left(z(1-z)Q^2\right) = \alpha_S(z(\theta), Q^2) \tag{3.10}$$

in agreement with most parton shower generators.

As color coherence is approximated by demanding that the angles of subsequent emissions decrease – the effect known as *angular ordering* as introduced in section 2.4 – the resulting phase space cuts have to be implemented in the parton shower. The opening angle[15] $\cos\vartheta$ is given by

$$\cos\vartheta = 1 - \frac{t}{2z(1-z)E^2} \tag{3.11}$$

in the approximation for massless children. Using $z(1-z) \leq \frac{1}{4}$ this can be used to give a cut on the scale of a next branching

$$t \leq E^2 \frac{1-\cos\vartheta_{cut}}{2} \tag{3.12}$$

for the branching to have an opening angle less than $\cos\vartheta_{cut}$. An additional cut on $z$ (see section B.2.2 for a derivation) is necessary:

$$\left|z - \frac{1}{2}\right| \leq \frac{\beta}{2}\sqrt{1 - \frac{t}{\beta^2 E^2}\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}}}. \tag{3.13}$$

With these phase space cuts angular ordering is enforced in the approximation of massless daughter partons. However the inclusion of this constraint

[15] not to be confused with the angle $\cos\theta$ used in the description of branchings

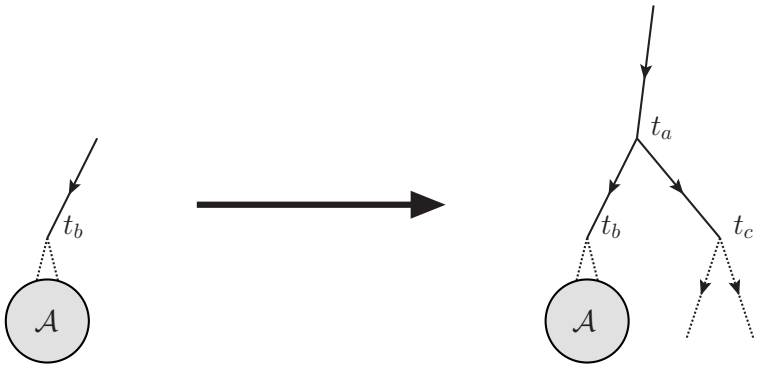


Figure 3.2: Schematic view of a double branching in ISR: Before the double branching (left): At the scale $t_b$ a parton $b$ exists. For this parton, the scale $t_a$ of the branching that produced this parton, the corresponding energy ratio $z_a$ and the scale $t_c$ where the emitted parton $c$ branches, and if necessary the remaining quantities ($v_c$) are simulated (on the right). The gray circle $\mathcal{A}$ stands for the remaining event, possibly including already found emissions in the initial state.

demands keeping track of the used energy $E$ and the used angle $\cos\vartheta_{cut}$ either by explicitly storing their values for every branching or by using a distinct rule to calculate them for every branching.

As a minor extension we allow for parton masses, although these are only taken into account when distributing momenta, the splitting functions are still taken for massless daughter partons.

## 3.3 Introducing the Analytic Initial-State Parton Shower

For physics at the LHC, a parton shower has to be able to describe both, initial- and final-state radiation. We therefore implement an initial-state parton shower satisfying the requirement of analyticity analogously to the parton shower for the final state. The algorithm for the analytic initial-state parton shower, the technical details, its implementation and validation are the main cornerstones of this thesis.

The changes applied to the final-state shower cannot be transferred to the initial-state shower[16]. A different set of changes is needed to reformulate

[16]Due to the negative virtualities all momenta and energies would be imaginary in the

the initial-state parton shower in order to fulfill the demand of analyticity. The known Sudakov factor for initial-state radiation is, repeated here from equation (2.24)

$$\Delta_b^{ISR}(t_a, t_b) = \exp\left[ -\int_{|t_a|}^{|t_b|} \mathrm{d}t' \int_0^1 \frac{\mathrm{d}z}{z} \frac{\alpha_S}{2\pi t'} \sum_{a,c} \left( P_{a\to bc}(z) + P_{a\to cb}(z) \right) \frac{f_a\left(\frac{x_b}{z}, t'\right)}{f_b(x_b, t')} \right] \quad (3.14)$$

with the splitting function $P_{a\to bc}(z)$ for a parton of type $a$ branching into two partons of type $b$ and $c$ and the parton distribution functions $f_a(x,t)$. The conservation of momentum can be enforced by explicitly vetoing momentum conservation-violating branchings directly in the Sudakov factor. To do this, the branching of the mother parton and the branching of the emitted parton have to be simulated simultaneously, cf. figure 3.2.

Otherwise, after the generation of the branching, the virtualities $t_a$ and $t_b$, and thus the absolute values of the momenta $|\mathbf{p}_a|$ and $|\mathbf{p}_b|$ would be fixed, while the virtuality $t_c$ and thus $|\mathbf{p}_c|$ would be unknown. Therefore it cannot be guaranteed that the subsequent simulation of the parton $c$ will not break the momentum conservation in the branching $a \to bc$.

The simulated branching therefore effectively becomes a $1 \to 2$ (if the emitted parton does not branch) or a $1 \to 3$ (if the emitted parton branches) branching. The Sudakov factor that takes the emitted parton's branching into account can be written in the form

$$\Delta_b^{ISR}(t_a, t_b) = \exp\left[ -\int_{|t_a|}^{|t_b|} \mathrm{d}t' \int_0^1 \frac{\mathrm{d}z}{z} \frac{\alpha_S}{2\pi t'} \sum_{a,c} \int_0^{t'} \mathrm{d}t_c \, \mathcal{P}_c(t_c| - t', z) \, \bar{\Theta}\left(-t', t_b, t_c, z_a, E_a\right) \left( P_{a\to bc}(z) + P_{a\to cb}(z) \right) \frac{f_a\left(\frac{x_b}{z}, t'\right)}{f_b(x_b, t')} \right] \quad (3.15)$$

with the veto function

$$\bar{\Theta}\big(t_a, t_b, t_c, z_a, E_a\big) = \Theta\big(|\mathbf{p}_b| + |\mathbf{p}_c| - |\mathbf{p}_a|\big) \cdot \Theta\big(|\mathbf{p}_a| - ||\mathbf{p}_b| - |\mathbf{p}_c||\big) \quad (3.16)$$

and the one-parton branching distribution function for the emitted parton $c$

$$\mathcal{P}_c(t_c| - t', z). \quad (3.17)$$

mothers restframe.

This function gives the probability distribution for the branching of the emitted timelike parton as a function of the branching this parton was produced in, described by $-t'$ and $z$. The veto function ensures that the three partons $a, b, c$ can be combined in a branching that conserves momentum by enforcing the triangle inequality. By adding more terms in $\bar{\Theta}$ it can also be used to impose cuts for angular ordering or a minimum energy for the emitted timelike parton. If the emitted parton branches, its final-state parton shower can now be simulated further by the use of the known double branching probabilities from the analytic final-state radiation.

However there is a slight difference in the interpretation of the known one-branching Sudakov factor as used for example in `PYTHIA` (cf. equation (3.14)) and the supplemented one in equation (3.15). In the former, the probability for a branching is independent of the available allowed branchings of the emitted parton, while in the latter the probability for a branching is reduced when the emitted parton has a restricted phase space for branchings. Therefore the supplemented Sudakov factor rather resembles a conditional probability, hence the notation in (3.17).

Using these prescriptions, double branch probability distributions can be formulated, analogously to the ones formulated for final-state radiation. The probability for no earlier branching, which means that the parton is directly emitted by the hadron and therefore assumed to be on-shell, $t_a \to m_a^2$, consists of the Sudakov factor $\Delta_b^{ISR}(-t_{cut}, t_b)$ and a $\delta$-distribution forcing the parton to be on-shell. It can thus be formulated in the form

$$\mathcal{P}_b^{nb}(t_a; t_b, t_{cut}) = \Delta_b^{ISR}(-t_{cut}, t_b)\delta\left(t_a - m_a^2\right). \tag{3.18}$$

In case an earlier branching is found, the common single branch probability would be

$$\begin{aligned}\mathcal{P}_{a\to bc}(t_a, z_a; t_b, t_{cut}) \;=\; & \frac{\alpha_S}{2\pi t_a}\frac{1}{z_a}P_{a\to bc}(z_a)\frac{f_a\left(x_a, t_a\right)}{f_b\left(x_b, t_a\right)} \\ & \cdot \Delta_b^{ISR}\left(t_a, t_b\right)\Theta\left(t_a - t_b\right)\Theta\left(-t_a - t_{cut}\right)\end{aligned} \tag{3.19}$$

with the Sudakov factor $\Delta_b^{ISR}(t_a, t_b)$, a relative weight given by the ratio of parton distribution functions, $\frac{f_a(x_a,t_a)}{f_b(x_b,t_a)}$, the probability for the branching itself, $\frac{\alpha_S}{2\pi t_a}\frac{1}{z_a}P_{a\to bc}(z_a)$ and two step functions that force the parton to be in the correct range of virtuality.

For the transition to analytic showers, a dependence on the scale of the branching of the emitted parton $t_c$ is introduced. Thus, two different cases have to be considered. In the case of the emitted parton $c$ not branching further, the corresponding probability distribution is supplemented by the

no-branching-probability $\mathcal{P}_c^{nb}$ for the emitted parton $c$ and the veto function $\bar{\Theta}$. It can be written in the way

$$\begin{aligned}
&\mathcal{P}^{br,nb}_{a\to bc}(t_a,t_c,z_a;t_b,t_{cut}) \\
&\quad= \frac{\alpha_S}{2\pi t_a}\bar{\Theta}\left(t_a,t_b,t_c,z_a,E_a\right)\frac{1}{z_a}P_{a\to bc}(z_a)\frac{f_a\left(x_a,t_a\right)}{f_b\left(x_b,t_a\right)} \\
&\qquad\cdot\Delta_b^{ISR}(t_a,t_b)\Theta\left(t_a-t_b\right)\Theta\left(-t_a-t_{cut}\right)\mathcal{P}_c^{nb}\left(t_c;|t_a|,t_{cut}\right). \qquad (3.20)
\end{aligned}$$

In the case in which the emitted parton undergoes another branching, the distribution is supplemented by the single branch probability for the emitted parton $\mathcal{P}^{br}_{c\to de}$ and has the form

$$\begin{aligned}
&\mathcal{P}^{br,br}_{a\to bc\to bde}(t_a,t_c,z_a,v_c;t_b,t_{cut}) \\
&\quad= \frac{\alpha_S}{2\pi t_a}\bar{\Theta}\left(t_a,t_b,t_c,z_a,E_a\right)\frac{1}{z_a}P_{a\to bc}(z_a)\frac{f_a\left(x_a,t_a\right)}{f_b\left(x_b,t_a\right)}\Delta_b^{ISR}(t_a,t_b) \\
&\qquad\cdot\Theta\left(t_a-t_b\right)\Theta\left(-t_a-t_{cut}\right)\mathcal{P}^{br}_{c\to de}\left(t_c,v_c;|t_a|,t_{cut}\right). \qquad (3.21)
\end{aligned}$$

Using these expressions, the probability distribution for the scale $t_a$ can therefore be written analogously to the case of final-state parton showers, equation (3.6), in the form

$$\begin{aligned}
\mathcal{P}_b(t_a;t_b,t_{cut}) &= \mathcal{P}_b^{nb}(t_a;t_b,t_{cut}) \\
&+ \sum_{a,c}\int \mathrm{d}z_a\int \mathrm{d}t_c\,\mathcal{P}^{br,nb}_{a\to bc}(t_a,t_c,z_a;t_b,t_{cut}) \\
&+ \sum_{a,c}\int \mathrm{d}z_a\int \mathrm{d}t_c\,\mathcal{P}^{br,nb}_{a\to cb}(t_a,t_c,z_a;t_b,t_{cut}) \\
&+ \sum_{a,c,d,e}\int \mathrm{d}z_a\int \mathrm{d}t_c\int \mathrm{d}v_c\,\mathcal{P}^{br,br}_{a\to bc\to bde}(t_a,t_c,z_a,v_c;t_b,t_{cut}) \\
&+ \sum_{a,c,d,e}\int \mathrm{d}z_a\int \mathrm{d}t_c\int \mathrm{d}v_c\,\mathcal{P}^{br,br}_{a\to cb\to deb}(t_a,t_c,z_a,v_c;t_b,t_{cut}). \qquad (3.22)
\end{aligned}$$

The probability distributions for the parton species and the energy fractions $z$ follow directly from this equation.

One aspect of initial-state parton showers that is a rather critical technical point, is the assignment of momenta for the first branching in the initial state. By the first branching we mean the branching closest to the hard interaction. As the partons in the hard matrix element are on the mass shell and often assumed to be massless, any branching would be kinematically forbidden. Therefore the partons have to be set off-shell in order to allow

for kinematically allowed branchings. This is done by simultaneously scaling the partons' momenta, until the invariant mass squared reaches the scale of the first branching,

$$t = p^2 = E^2 - \mathbf{p}^2 = t_{first} \ll 0.$$

The distribution of $t_{first}$ is obtained by solving a Sudakov factor similar to the one given in equation (3.15), but with the terms corresponding to the emitted parton removed,

$$\Delta_b^{ISR}(t_{first}, t_b) = \exp\left[-\int_{|t_a|}^{|t_b|} \mathrm{d}t' \int_0^1 \frac{\mathrm{d}z}{z} \frac{\alpha_S}{2\pi t'} \sum_{a,c} \left( P_{a\to bc}(z) + P_{a\to cb}(z) \right) \frac{f_a\left(\frac{x_b}{z}, t'\right)}{f_b(x_b, t')} \right]. \quad (3.23)$$

Thus the initial-state parton shower is not started from the two partons in the initial state of the matrix element, but from copies of them that have their momenta assigned in the following way:

$$t_1 = t_{first\,1} \qquad t_2 = t_{first\,2} \quad (3.24)$$

$$E_1 = \frac{\hat{s} + t_1 - t_2}{4\sqrt{\hat{s}}} \qquad E_2 = \frac{\sqrt{\hat{s}}}{2} - E_1 \quad (3.25)$$

$$|\mathbf{p}_1| = |\mathbf{p}_2| = \sqrt{E_1^2 - t_1} \qquad \mathbf{p}_1 = -\mathbf{p}_2 \quad (3.26)$$

By doing so, both partons are set off-shell so that branchings are kinematically allowed, while conserving the total energy and momentum. Another possibility would be to enlarge the three-momenta so that the scales are equal to the negative partonic center-of-mass energy, $t \to -\hat{s}^2$, and then start the shower from there, but this starting configuration has the disadvantage that the three-momenta of the initial partons tend to be very large, so that it becomes very hard to find kinematically allowed branchings.

## 3.4 On the Choice of the Evolution Variable

There has been a trend towards transverse-momentum- or $p_T$-ordered parton showers during the last years, mainly driven by the easier implementation of angular ordering in $p_T$-ordered showers. Our algorithm for analytic parton showers retains virtuality $t$ as ordering variable. This is because the formulation of a $p_T$-ordered analytic parton shower led to a cyclic dependence in the variables as follows:

The integrand in the Sudakov-factor depends on the two variables energy ratio $z$ and virtuality $t$ (and some more variables that are dropped for now), we denote the integrand here by $I_{tz}(t, z)$. Replacing the energy ratio $z$ by the angle $\cos\theta$ (cf. equation (3.7)) leads to an integrand of the form $I_{t\theta}(t, \theta) = I_{tz}(t, z(\theta, t))$. The relation between virtuality and transverse momentum was given in (2.4), $t = \frac{\mathbf{p}_\perp^2}{z(1-z)}$. To replace the virtuality $t$ by the transverse momentum $p_T$, one would use the formula

$$t = \frac{\mathbf{p}_\perp^2}{z(1-z)} = \frac{\mathbf{p}_\perp^2}{z(\theta, t)(1 - z(\theta, t))}, \tag{3.27}$$

thus introducing a circular dependence due to the replacement of the energy ratio $z$ by the angle $\theta$. Due to this problem, the ansatz of an analytic $p_T$-ordered parton shower was not pursued. It is unclear whether this problem persists for any reformulation of the parton shower. The search for a $p_T$-ordered analytic parton shower is one possible future task.

# 4 Implementation

In order to show the practical feasibility, we implemented the algorithms for the analytic parton showers in the form of `Fortran90/03` modules. The main focus was to build an extension to the event generator `WHIZARD`. The implementation in the form of `Fortran` modules allowed to pursue the basic design patterns from `WHIZARD`. Among these are a high level of modularization and the use of object-oriented like features of the language in order to guarantee a clear structure and good maintenance of the code. Despite the integration in `WHIZARD`, the shower modules are self-contained and can be used from a standalone program as well[17]. We connected the event generator `WHIZARD` to the parton shower modules by means of a well-separated interface. Moreover, in order to generate more realistic events, we implemented a version of the MLM matching scheme presented in section 2.9 together with an interface between `WHIZARD` and the matching procedure. In addition, we attached the `Fortran77` version of `PYTHIA` to the main `WHIZARD` core by another interface to use its hadronization routines. We then extended this interface to be also able to use `PYTHIA`'s parton shower routines from within the structure of `WHIZARD`, instead of the internal analytic parton shower.

This chapter is structured as follows: First we will give an overview of the technical details of the implementation of the analytic parton shower in section 4.1. The prerequisites for a standalone compilation of the analytic parton shower modules are given in section 4.2. Then we discuss shortly the event generator `WHIZARD` and its structure in section 4.3. Section 4.4 then describes the interface implemented to use the parton shower from `WHIZARD`. The implementation of our variant of the MLM matching scheme is subject of section 4.5. The part in section 4.6 describes the interfaces between `WHIZARD` and `PYTHIA`, namely one interface to substitute the analytic parton shower with `PYTHIA`'s parton shower and one interface to use the hadronization routines from `PYTHIA`.

## 4.1 Implementation of the Analytic Parton Shower

In the current version of our code, three different parton shower algorithms are implemented:

- a virtuality-ordered analytic parton shower for final-state radiation as presented in section 3.2,
- a virtuality-ordered analytic parton shower for initial-state radiation as presented in section 3.3,

[17] along with the inclusion of some auxiliary modules from `WHIZARD`

- a $p_T$-ordered shower for initial-state radiation.

The $p_T$-ordered parton shower is a simplified reproduction of the $p_T$-ordered initial-state parton shower in PYTHIA and was only implemented for comparison during an early stage of the implementation. It will not be discussed further in this chapter.

### 4.1.1 Structure of the Source Code

The source code is located in the directory src/shower in WHIZARD's source code repository. The directory consists of the following files:

- shower_basics_module contains the basic settings for the shower and some basic routines, e.g. for the calculation of the running of the coupling constant $\alpha_S$.

- shower_parton_module contains the description of the properties of single partons. Most of the file's content is used for book-keeping purposes, e.g. assigning momenta to daughter particles once a branching has been fully simulated.

- shower_module is the main module for the shower. The general data types are implemented in this file, as are the most important procedures for the algorithms.

- shower_topythia contains an interface to PYTHIA. It allows the content of an event to be written into PYTHIA's data structures, mimicking an event showered by PYTHIA. The event can then be hadronized by PYTHIA's hadronization routines.

- shower_dummy contains dummy copies for all routines and variables used by WHIZARD. This file is included to substitute for the real implementation in case WHIZARD is to be compiled without the shower. If any of the procedures is called, e.g. by requesting showered events to be generated but using a WHIZARD binary that has been compiled without the shower, it terminates execution with an error message.

Moreover, some basic modules from WHIZARD are used in the source code and can hence be considered as part of the implementation, see section 4.2 for a list of these modules.

### 4.1.2 Fundamental Data Structure

The three main data types used by the parton shower are `parton_t` for single partons, `shower_interaction_t` for interactions generated by matrix elements and `shower_t` as the general type. The two former types are accompanied by types encapsulating pointers to these types, `parton_pointer_t` and `shower_interaction_pointer_t`[18].

The information for each parton is stored in a variable of type `parton_t`, defined in `shower_parton_module.f90`.

```
type :: parton_t
  integer :: nr=0
  integer :: typ=0
  type(vector4_t) :: momentum = vector4_null
  real(default) :: t  = 0._default
  real(default) :: scale = 0._default
  real(default) :: z = 0._default
  real(default) :: costheta = 0._default
  real(default) :: x=0._default
  logical :: simulated=.false.
  logical :: belongstoFSR=.true.
  logical :: belongstointeraction=.false.
  type(parton_t), pointer :: parent => null ()
  type(parton_t), pointer :: child1 => null ()
  type(parton_t), pointer :: child2 => null ()
  type(parton_t), pointer :: initial => null ()
  integer :: c1 = 0, c2 = 0
  integer :: interactionnr = 0
end type parton_t
```

The information stored for each parton can be grouped as follows

- general properties:
  - `typ` type of the particle, given as a PDG-Code [82], e.g. 1 for a *down* quark, $-1$ for an *antidown* quark and 21 for a gluon,
- kinematical properties:
  - `momentum` four-momentum of the parton,
  - `t` the four-momentum squared,

[18] This is needed due to `Fortran`'s incapability to build arrays of pointers.

  - `scale` the scale of the particle, equal to the virtuality $t$ for partons in a virtuality-ordered shower,

- book-keeping information:
  - `nr` each particle is assigned a number that can serve as an identifier,
  - `interactionnr` gives the number of the interaction this parton is connected to, as stored in the `interactions` array in `shower_t` (see below),
  - `simulated` whether this particle has already been simulated,
  - `belongstoFSR` to distinguish between partons belonging to the initial and final-state parton shower,
  - `belongstointeraction` to distinguish partons belonging to an interaction described by a matrix element,
  - `c1`, `c2` color and anticolor indices in the $N_C \to \infty$ limit[19],

- information about this parton's branchings:
  - `parent` a pointer to the parton that emitted the current parton (if any),
  - `child1`, `child2` pointers to the two partons this parton branches into (if any),
  - `z` the energy ratio of the first daughter and the current parton,
  - `costheta` the generated $\cos\theta$ value for this parton's branching,

- special information for partons in the initial-state parton shower:
  - `x` this parton's momentum fraction,
  - `initial` is a pointer to the parton's initial hadron.

A dedicated subroutine, `shower_add_interaction2ton`, is implemented to add a hard $2 \to n$ interaction. It takes an array of the partons involved in the interaction as its argument. It is assumed that the first and second particle in the list are the incoming particles, the remaining particles are assumed to be the final particles. It is assumed that there are at least two

[19] Parton showers work in the *large* $N_C$ limit, where the number of different color charges is taken to be infinite. In this approximation, a gluon can be treated to be a composition of one color and one anticolor charge and interferences between different color structures are disregarded.

outgoing particles. Each variable of the `shower_interaction_t` type consists of an array of the incoming and outgoing partons. The data type is called `shower_interaction_t` to avoid confusion with the `interaction_t` type built-in within `WHIZARD`.

```
type :: shower_interaction_t
   type(parton_pointer_t), dimension(:), allocatable :: partons
end type shower_interaction_t
```

The main data type is the `shower_t` type. It is intended to represent one event. The complete event information is stored in a variable of this type.

```
type :: shower_t
   type(interaction_pointer_t), dimension(:),
        allocatable :: interactions
   type(parton_pointer_t), dimension(:),
        allocatable :: partons
   integer :: next_free_nr
   integer :: next_color_nr
   logical :: valid
end type shower_t
```

The `interactions` array holds pointers to the interactions described by matrix elements. As currently only one interaction per event is generated by `WHIZARD`, a single pointer would be sufficient. In preparation for the implementation of multiple interactions, we chose to use an array instead. The `partons` array serves as a collection of all partons present in the event, regardless of their state. Note that the array is not the only way to access the partons, in addition all partons are part of trees using the `parent` and `child` pointers in `parton_t`. Thus partons can be accessed by trees either starting at one member of the `partons` array or at a parton that is part of an interaction stored in the `interactions` array. The subroutine `shower_add_children` is implemented to consistently add new partons to the tree and the array, respecting all parent and child connections. The integers `next_free_nr` and `next_color_nr` are used to distribute the numbers and color numbers to the partons. The boolean variable `valid` is used to indicate that no error occurred in the processing of the event.

The variable is initialized by a call to `shower_create` and destroyed by a call to `shower_final`. Procedures are provided that add interactions, perform the initial- and final-state parton showers and print the event to the screen or an Les Houches event file (LHEF)[83].

### 4.1.3 Implementation of the Analytic FSR

Some arrangements for the final-state parton shower are done while adding the interaction to the shower record. Because of four-momentum conservation the parton shower cannot be started from single partons[20]. Therefore the outgoing partons are clustered until only a minimum number of partons or *pseudo-partons* remains. A pseudo-parton is an imaginary parton that is assigned a momentum as well as parent/child-connections to other partons or pseudo-partons, but not assigned a well-defined type. The resulting tree of partons and pseudo-partons is called the *pseudo-parton shower history*. The clustering is done using the known $k_T$-algorithm[21]. This means that for every combination of particles in the final state the value

$$y_{ij} = \frac{2\min(E_i, E_j)}{s}(1 - \cos\theta_{ij}) \tag{4.1}$$

is calculated, then the pair of partons with the lowest value of $y_{ij}$ is replaced by one pseudo-parton with combined momentum. By doing this, the intermediate partons are automatically put off-shell, so that the branching can be understood as having taken place at a scale corresponding to the off-shellness. In the current status of the implementation, the clustering is done regardless of whether the clustering is allowed. A procedure that is intended to check if the clustering is allowed is present, but not completely implemented at the moment. As an example, in a $q\bar{q} \to \gamma \to q\bar{q}g$ process, the gluon has to be clustered with one of the quarks first to generate a parton shower history $\gamma \to q\bar{q} \to q\bar{q}g$, while the clustering of the two quarks first is forbidden as it would give a parton shower history like $\gamma \to gC \to q\bar{q}g$ with $C$ either being a gluon, photon or a $Z$-Boson. On the other hand a process $q\bar{q} \to g \to q'\bar{q}'g$ would allow all possible clusterings.

During the simulation of the parton shower, the actual branchings are generated by searching for the parton with the highest virtuality and yet unsimulated daughters. For both daughters the scale of the next branching

[20] This can be seen from a $q \to qg$ branching in the initial quark's rest frame. The four-momenta are given by

$$p_{qi} = p_{qf} + p_g$$

and therefore

$$m_q^2 = p_{qi}^2 = (p_{qf} + p_g)^2 = m_q^2 + 2p_{qf} \cdot p_g.$$

As $p_{qf} \cdot p_g > 0$ this equation can never be fulfilled. This is avoided by setting the initial parton off-shell. If there is only one parton in the initial state, this obviously violates momentum conservation. In the case of more partons in the initial state, the additional momenta can be chosen to be balanced.

[21] See A.2 for an introduction to the $k_T$-algorithm.

is simulated. The actual simulation of the double branchings is done by the subroutine `shower_simulate_children_ana`. This procedure is repeatedly called until all particles are simulated. As it is assumed that there are no interferences between the FSRs of different hard interactions, the FSRs for each interaction are generated independently. For this purpose, the subroutine `shower_interaction_generate_fsr2ton` is called for each interaction.

The integrals over $\cos\theta$[22] in the Sudakov factors (see e.g. (3.9)) are evaluated in the subroutine `parton_simulate_stept`. The domain of integration is divided into bins of variable width. The bin width is chosen to be small in regions with possible divergences. As the possible divergences are of type $\sim \frac{1}{1+c}$ or $\sim \frac{1}{1-c}$, a function of the form $x_1 + \frac{x_2}{1+c} + \frac{x_3}{1-c}$ is fitted to the values of the integrand at the borders and in the middle of the bin. Integration over the fitted function can be achieved analytically, yielding the integral for one bin

$$I_{bin} = x_1 \Delta c + x_2 \log\left(\frac{1+c+\Delta c}{1+c}\right) - x_3 \log\left(\frac{1-c-\Delta c}{1-c}\right)$$

with $c$ as the value of $\cos\theta$ at the left boundary of the bin and the bin width $\Delta c$.

### 4.1.4 Implementation of the Analytic ISR

We chose the structure of the implementation to be easily extendable to include interleaved evolution[23]. Thus the task of simulating a branching in the initial-state parton shower is divided into two procedures, one that calculates the scale of the next branching and one that actually executes the found branching. These two procedures are `shower_generate_next_isr_branching` and `shower_execute_next_isr_branching`. The first function returns a pointer to the next branching parton. This pointer is later used by `shower_execute_next_isr_branching` to execute the branching. This is repeated as long as `shower_generate_next_isr_branching` finds partons that undergo a branching.

The simulation of the initial-state parton shower is done by a loop of the form

```
do
  pp=shower_generate_next_isr_branching(shower)

  if(.not. associated(pp%p)) then
    exit
```

[22] For brevity, we use the abbreviation $c = \cos\theta$.

[23] See section 1.4.2 for a short introduction to interleaved evolution.

```
    end if

    call shower_execute_next_isr_branching(shower, pp)
  end do
```

The expansion to include interleaved evolution is straight-forward. Instead of only taking into account a possible next branching in the parton shower via the subroutine `shower_generate_next_isr_branching`, the scale of a possible next multiple interaction is evaluated as well. Then, depending on which of these happen at a higher scale, either the additional interaction or the additional branching is executed. If the procedures did not find an additional interaction nor an additional branching, the loop is ended. The expanded form of the loop would then look like[24]

```
do
  next_brancher=shower_generate_next_isr_branching(shower)
  next_int_scale=get_next_mi_scale()

  if(next_brancher%p==null .and. next_int_scale==0) then
    exit
  end if

  if(next_brancher%p%scale < next_int_scale) then
    call execute_next_interaction()
  else
    call shower_execute_next_isr_branching(shower, pp)
  end if
end do
```

After the loop has ended by finding no more branchings (and finding no more multiple interactions in the interleaved case), the subroutine `shower_generate_fsr_for_partons_emitted_in_ISR` is called. It simulates the final-state parton showers originating from the partons emitted in the initial-state radiation. Finally, calls to three subroutines needed for book-keeping are made. Firstly, the call to `shower_boost_to_labframe` is necessary because the evolution of the initial-state parton shower is always performed in the center-of-mass frame of the two most primal partons, thus leading to Lorentz boosts for every new branching. The final boost is then needed to transform the event back to the lab frame. The two latter procedures, `shower_generate_primordial_kt` and then finally `shower_update_`

[24] This has not yet been implemented. Therefore, the added procedure names do not correspond to actual procedures.

`beamremnants`, implement the treatment of beam remnants as presented in the next section.

### 4.1.5 Beam Remnants

We implemented a very rudimentary treatment of beam remnants, with the main purpose of being able to provide a color-neutral input to the hadronization. In dependence of the emitted particle the beam remnant is assumed to consist of one or two partons. The procedure for determining these partons' flavors and momenta is given below.

The given procedure obviously only applies in the case of only one emitted parton per proton, that is in the case of only one hard interaction. As an implementation of an interleaved multiple interactions/initial-state radiation evolution along the lines of the *Interleaved Evolution* approach [36] is in preparation, this simple treatment of beam remnants will become inapplicable. Thus a more sophisticated treatment, including a treatment of multiple emitted partons, will be implemented in the future.

**Primordial $k_T$** In contrast to the formulation of PDFs and the DGLAP equations in the infinite-momentum frame and thus omitting any transverse momentum, experimental results can be better reproduced assuming a non-vanishing transverse momentum of the initial partons. The beam remnant will act as a recoiler for the initial partons. This transverse momentum is commonly referred to as *primordial* $k_T$. There is no theoretical description for the distribution of the primordial $k_T$. The distribution is usually modeled using a Gaussian distribution with an additional constraint $|k_T| < k_{T\,max}$. This $k_{T\,max}$ and the width of the Gaussian introduce two additional parameters that have to be fitted using experimental data. The extension to other distributions is as well possible.

We implemented the Gaussian distribution. The implementation is available via the subroutine `shower_apply_primordial_kt`. This subroutine will generate the primordial $k_T$ for both initial partons and apply them using a random polar angle. It will also perform the Lorentz boost necessitated by the change of the initial partons' momenta.

**Flavors** The flavors of the beam remnant are chosen according to a simplified procedure from PYTHIA [23, section 11.1.1]. Depending on the flavor of the emitted parton the flavors of the beam remnant are chosen (These rules apply for protons as the initial hadrons, with obvious alterations for antiprotons.):

- A valence quark of the hadron is assumed to leave behind a diquark beam remnant. A $ud$-diquark is assumed to be a $ud_1$ in 25% and a $ud_0$ in 75% of the cases, while a $uu$-diquark is always a $uu_1$.

- A gluon is assumed to leave behind a color octet state, that is divided into a color triplet quark and an anticolor triplet antiquark. The division is 1/6 into $u + ud_1$, 1/2 into $u + ud_0$ and 1/3 into $d + uu_1$.

- A sea quark, such as an $s$, leaves behind an $uud\bar{s}$ state, that is subdivided into a meson and a diquark. The relative probabilities are 1/6 into $u\bar{s} + ud_1$, 1/2 into $u\bar{s} + ud_0$ and 1/3 into $d\bar{s} + uu_1$.

- An antiquark $\bar{q}$ leaves behind a $uudq$ state, that is divided into a baryon and a quark. Since mostly the $q\bar{q}$ pair comes from an emission of a gluon, the subdivision $uud + q$ is not allowed as it would correspond to a color singlet gluon. The subdivision is therefore 2/3 into $udq + u$ and 1/3 into $uuq + d$. The three quark state $uuq$ or $udq$ is then replaced by the corresponding baryon of lowest spin.

This obviously only applies in the case of only one emitted parton, that is the case of only one hard interaction. For the case of multiple interactions a more sophisticated approach has to be found.

**Momenta** The total momentum of the beam remnant is given by the remaining momentum of the hadron after the emitted particle has been removed. In case the beam remnant consists of only one parton, this parton is assigned the complete momentum. Otherwise, if the beam remnant consists of a diquark and a quark the momentum is distributed in equal parts to the both constituents. If the beam remnant consists of two constituents with one of them being a meson or baryon, the energy is distributed in equal parts but the three-momentum is distributed so that the hadron is on-shell and the quark is assigned the remaining momentum. This procedure generates on-shell colorless particles and off-shell colored particles. The colored particles' off-shellness is absorbed in the hadronization.

## 4.2 Standalone Compilation

When compiling the source code located in the `src/shower` directory in order to generate a standalone program, two additional components are required. First, the shower demands the inclusion of the LHAPDF [30] library. As parton distribution functions are needed for initial-state radiation, a version

compiled without the LHAPDF library would only be applicable to final-state radiation. For now, we do not allow for compilation without LHAPDF, but this could easily be allowed, e.g. for use at a lepton collider[25]. Second, for simplicity, the parton shower code uses some basic components from WHIZARD, e.g. the kinds module that defines the precision of floating point variables or the lorentz module that defines four-vectors, Lorentz boosts and their algebra. For the standalone compilation the needed modules have to be provided.

If both these requirements are met, a command line invoking the compilation could then look like

```
gfortran -lLHAPDF -L<...LHAPDF directory...>
     constants.f90  iso_varying_string.f90 limits.f90
     file_utils.f90 lorentz.f90 tao_random_numbers.f90
     shower_basics_module.f90 shower_parton_module.f90
     shower_module.f90 main.f90
```

with the main program in main.f90. This main program must provide the matrix-element events, for example by simulating them or reading them from an event file. A very simple example main program assuming a constant matrix element is given in appendix C.1.

## 4.3 Structure of WHIZARD

WHIZARD is an event generator for a variety of applications at hadron, lepton and photon colliders. To achieve this, it has been supplemented by several subprograms. The name WHIZARD stands for the whole package as well as for the core program without any subprograms. The matrix-element generator O'Mega [55] and the integration library VAMP [84] are the two most important subprograms in the WHIZARD package. Both, VAMP and O'Mega, are self-contained packages and can also be used without WHIZARD.

Technically, WHIZARD consists of five libraries:

- libaux
- libwhizard-core
- libvamp
- libmodels

[25] One possible way to compile the shower without the need for the LHAPDF libraries would be to use the LHAPDF dummy included in the WHIZARD package in the directory src/lhapdf.

- libomega

The libaux library contains basic auxiliary modules needed by more than one other module, for example all WHIZARD modules needed for the standalone parton shower program, except for the random number generator module in tao_random_numbers.f90 that is provided by VAMP. The libraries libvamp, libomega and libmodels are provided by the subprograms VAMP and O'Mega, respectively.

The library libwhizard-core contains all modules needed to read all input into WHIZARD, steer O'Mega and VAMP, manage the phase space integration, generate and analyze events and write out results. The source code for this library resides in the file src/whizard-core/whizard.nw, a noweb file that contains both, source code and documentation.

For a more elaborate description of WHIZARD and its structure, see the recent WHIZARD publication [29].

## 4.4 The Interface between WHIZARD and the Parton Shower

We implemented the parton shower interface as an additional module shower_interface[26] in the library libwhizard-core. The source code is located in the file src/whizard-core/whizard.nw like most other modules from that library. The module's main procedure is apply_shower_particle_set. A particle_set is a component of WHIZARD's type for events, it basically consists of a list of particles. During the event generation, the procedure apply_shower_particle_set is called if necessary. Apart from the implementation of matching, see section 4.5, the main task of this procedure is to call one of the procedures for showering, apply_PYTHIAshower_particle_set and apply_WHIZARDshower_particle_set, depending on which parton shower was chosen, and to call the procedure for hadronization using PYTHIA, apply_PYTHIAhadronization. All these procedures are only called if activated in the WHIZARD input file.

Although they are two separate subroutines, the two procedures for showering share most of the steps. First, the event information from the particle set is transferred to the data structure of the respective parton shower. For WHIZARD's shower this is done by constructing an array holding all particles of the hard interaction and adding this to the shower data structure

[26] The name shower_interface is a misleading one. The module was originally intended as an interface to the parton shower only. However, the module outgrew its original purpose as the module now contains not only the interface to the shower but also to the hadronization, matching, etc. as well.

via calling shower_add_interaction2ton. To transfer the event to PYTHIA, the event information is written to a Les Houches Event File [83]. The file is not written to the hard disk drive, but only to a Fortran scratch file. This ensures that the data is kept in the memory all the time, thus minimizing the performance penalty induced due to the conversion of the event file. Apart from the particle list and all values that can be inferred from the list, the event file will only contain dummy information. However, the omitted information is irrelevant for the showering of the event. PYTHIA is configured to read in the Les Houches Event File and only perform the parton shower[27]. The simulation of the shower then reduces to a call to PYTHIA's function PYEVNT. For WHIZARD's internal shower, the actual showering is implemented by calling the procedures shower_generate_next_isr_branching, shower_execute_next_isr_branching and shower_interaction_generate_fsr2ton and some auxiliary functions in a meaningful combination. In a final step, the showered events are read back into WHIZARD. For both parton shower implementations, the showered event is written again to a Les Houches Event File, which is then added to WHIZARD's particle set via the procedure shower_add_lhef_to_particle_set. One additional step in apply_WHIZARDshower_particle_set is that if hadronization is switched on, the showered event is automatically written to PYTHIA's data structure via the use of the shower_topythia module. This is done because the hadronization routine assumes that the event information is already present in the PYTHIA's data structure. This assumption is included to avoid redundant transfers of particle information to PYTHIA in case PYTHIA is used for both, shower and hadronization. The hadronization procedure apply_PYTHIAhadronization then basically consists of a call to the PYEXEC routine from PYTHIA[28]. Like after the showering, the resulting event is written to a Les Houches Event File, that is then added to the original particle set. See sections 4.6.1 and 4.6.2 for more elaborate discussions of the interface to PYTHIA's parton shower and hadronization routines.

The reason for dividing the simulation in different steps for the shower and the hadronization is to be able to access the showered but not yet hadronized event. This information is needed for the MLM matching algorithm, as shown in the next section.

[27] By default, PYTHIA also simulates multiple interactions when simulating the initial-state parton shower. For a correct comparison between PYTHIA's parton shower and the built-in one, PYTHIA's treatment of multiple interactions would have to be manually disabled.

[28] Similar to the case for initial-state parton showers, the simulation of hadronization via PYTHIA also includes the simulation of hadronic decays unless explicitly switched off.

## 4.5 Implementation of MLM Matching

In the process of implementing the analytic parton shower, we also implemented a matching procedure according to the principles of the MLM approach in WHIZARD with the use of a $k_T$-clustering jet algorithm from the KTCLUS clustering package [85] (cf. section A.2). As MLM matching is not unambiguously defined in the literature, a freedom in the implementation remains. This freedom consists of the choice of jet definitions and a procedure to correlate jets described by the matrix elements with jets after the parton shower. We chose an implementation that was inspired by the discussion of the MLM matching in [74].

The steps in detail are:

1. The cross sections for the main process and processes with up to $N$ additional partons in the hard matrix element are calculated. The phase space has to satisfy the additional cuts enforced by the matching procedure

$$p_T > p_{T\,min}, \qquad |\eta| < |\eta_{max}|, \qquad \Delta R_{jj} > R_{min} \tag{4.2}$$

   with the transverse momentum $p_T$, the pseudo-rapidity $\eta$ and the $\eta-\phi$-distance between two jets $\Delta R_{jj}$. The values $p_{T\,min}$, $\eta_{max}$ and $R_{min}$ are set in the WHIZARD input file.

2. According to the relative probability $P(i)$ given by the relative size of the corresponding cross sections,

$$P(i) = \frac{\sigma_i}{\sum_j \sigma_j} \tag{4.3}$$

   a matrix-element event with $i$ additional partons is generated.

3. The $k_T$-clustering jet algorithm is applied to the final colored partons from the matrix element[29].

4. These events are then showered. This is done with WHIZARD's shower by default. For the purpose of comparison, one can replace the analytic shower by PYTHIA's parton shower without changing the rest of the calculation.

[29] Normally, the number of jets after the matrix element will be given by the number of colored outgoing partons. Recent studies of compressed SUSY spectra, these are SUSY spectra that contain nearly mass-degenerate particles, include very low energetic quarks as final partons in SUSY decay chains. These quarks will mostly be unable to initiate a jet and thus should not be included in the MLM matching. A clustering algorithm applied directly to the event as described by the matrix element will reject these low-energetic quarks.

5. After the shower evolution, the $k_T$-clustering jet algorithm is applied to the showered, but not yet hadronized event, taking only colored partons with a pseudo-rapidity $|\eta| < \eta_{max\,clus}$ into account. Jets are defined by a minimum jet-jet separation $y_{cut}$[30],

$$\eta_{max\,clus} = \eta_{clusfactor} * \eta_{max}, \tag{4.4}$$

$$y_{cut} = \Big[p_{T\,min} + \max\left(E_{T\,clusminE},\right. \\ \left. E_{T\,clusfactor} * p_{T\,min}\right)\Big]^2. \tag{4.5}$$

   The factors and hence the clustering variables can be varied as part of the systematics assessment, the defaults for these factors are chosen to be 1.

6. If the number of jets after the parton shower undershoots the number of jets after the matrix element, the event is discarded. When the number of jets after the parton shower overshoots the number of jets after the matrix element, the event is rejected as well, unless the number of matrix-element jets is equal to the maximum allowed number of matrix-element jets. In that case the scale $y_{cut}$ is adapted such that the number of reconstructed jets is reduced to the number of matrix-element jets, i.e. the jet resolution is lowered accordingly.

7. Then it is tested if the reconstructed jets match the matrix-element partons. This is done in an iterative way: The clustering is reapplied to a set consisting of the reconstructed jets and one of the matrix-element jets. Using this set, the jets at the jet resolution scale $y_{cut}$ are reconstructed. If the additional jet leads to an additional jet at the scale $y_{cut}$, the matrix-element jet is assumed not to be matched to any of the reconstructed jets after the parton shower and the event is discarded. Otherwise the matched reconstructed jet is removed from the set and this procedure is repeated for the next matrix-element jet. If and only if all matrix-element partons can be matched in this way, the event is accepted.

These steps are implemented in a `Fortran` module in the file `src/shower/mlm_matching_module.f90`. The main procedure of this module is simply called `mlm_matching`. Its input consists of a list of the particles' momenta as a described by the matrix element and a list of the momenta of the particles after the shower. The procedure constructs the respective jets from these

[30] see section A.2 for a short introduction to jet clustering

momenta and performs the steps 3,5,6 and 7 of the listing above. The first two steps and the parton shower are called from within WHIZARD.

## 4.6 Additional Interfaces

The interface to PYTHIA's parton shower was implemented during the development of the analytic parton shower to have a means of comparison. Nevertheless, it also provides users of WHIZARD with a way to generate showered events without interfacing external programs. As long as the analytic shower has not been validated using experimental data, it is recommended to use the PYTHIA shower for production runs. Then, it was only a minor task to also include an interface to PYTHIA's hadronization routines.

### 4.6.1 Interface to PYTHIA's Parton Shower

The subroutine incorporating the interface to PYTHIA's parton shower, apply_PYTHIAshower_particle_set, is called in the same way as the procedure for the internal shower. The body of the procedure basically consists of five steps:

- The passed event information is written to a LHEF. As for the interface to WHIZARD's parton shower, the file is not written to the hard disk drive, but only to a Fortran scratch file.
- PYTHIA is configured. This includes the setup for reading the event file, the transfer of settings from WHIZARD and the transfer of special settings for PYTHIA via PYTHIA's PYGIVE command.
- The actual showering is performed via a call to the PYTHIA routine PYEVNT.
- The showered event is retransferred to WHIZARD. This transfer is also done using an LHEF stored in a scratch file.
- If matching is enabled, the momenta of the partons belonging to the hard interaction are extracted from PYTHIA's data structure and saved for usage in the matching. The information has to be extracted from PYTHIA's data structure to respect Lorentz boosts made during the evolution of initial-state radiation.

### 4.6.2 Hadronization Interface

The hadronization routine closely resembles the routine for the interface to PYTHIA's parton shower. The main difference is that the call to PYEVNT is

replaced by a call to PYTHIA's hadronization routine PYEXEC. A detailed description of the usage of the interface will be given in the WHIZARD manual once the analytic parton shower is released as an official part of the WHIZARD package.

# 5 Validation and Results

We compared the predictions for the process $e^+e^- \to hadrons$ at LEP with center-of-mass energies of $\sqrt{s} = 91\,\text{GeV}$ and $\sqrt{s} = 133\,\text{GeV}$ and for $Z$ production, $p\bar{p}/pp \to Z + X$, at the Tevatron and the LHC at energies of $\sqrt{s} = 1.96\,\text{TeV}$ and $\sqrt{s} = 7\,\text{TeV}$, respectively. All event sets were generated using `WHIZARD`, which means the hard interaction was simulated by `WHIZARD/O'Mega`, the parton shower was either simulated using `PYTHIA`'s virtuality-ordered shower or `WHIZARD`'s own analytic shower, denoted in the plots by either `PYTHIA` PS or `WHIZARD` PS. For `WHIZARD`'s parton shower we used a first-order running $\alpha_S$ in the $\overline{\text{MS}}$-scheme, given by

$$\alpha_S\left(Q^2\right) = \frac{4\pi}{\left(11 - \frac{2}{3}n_f\right)\log\left(Q^2/\Lambda^2\right)}, \tag{5.1}$$

taking $n_f$ and $\Lambda$ as constants, neglecting the influence of flavor thresholds for the moment. For `PYTHIA`'s parton shower we used the same $\Lambda$ and $n_f$ values for a first-oder running $\alpha_S$, but with threshold effects enabled. If not stated otherwise, the values for $\Lambda$ were chosen to be $\Lambda = 0.19\,\text{GeV}$ for `WHIZARD`'s parton shower and $\Lambda = 0.29\,\text{GeV}$ for `PYTHIA`'s parton shower. The $\Lambda$ value for `WHIZARD` was chosen by hand to improve agreement with `PYTHIA`'s distributions. The hadronization, if activated, was simulated using `PYTHIA` with the hadronization tune from an analysis by the DELPHI collaboration [86, table 10, Dec. 93]. This tune was of course made using `PYTHIA`'s parton shower, but will be used in here together with `WHIZARD`'s parton shower as well. As the hadronization tune depends on the parton shower, using a tuning obtained with a different parton shower can lead to unsubstantial deviations in the results. As there is no tuning with `WHIZARD`'s parton shower available, we cannot give an estimate for the deviations. The possible tuning of our shower is beyond the scope of this thesis, presenting merely the algorithm, and will be left for future work. Given the fact that no tuning has been done, the shower already describes data in a QCD environment reasonably well.

The definitions of all observables are given in section A.1 in the appendix.

## 5.1 Final-State Radiation at Parton Level

Figures 5.1 to 5.4 show a comparison of distributions of event shapes at parton level for the process $e^+e^- \to u\bar{u}$ at a center-of-mass energy of $\sqrt{s} = 133\,\text{GeV}$. For both parton shower programs the events were showered with a cut-off virtuality $t_{cut} = 1\,\text{GeV}^2$, hadronization was disabled. The plots for thrust, thrust major and thrust minor show that `WHIZARD`'s parton shower

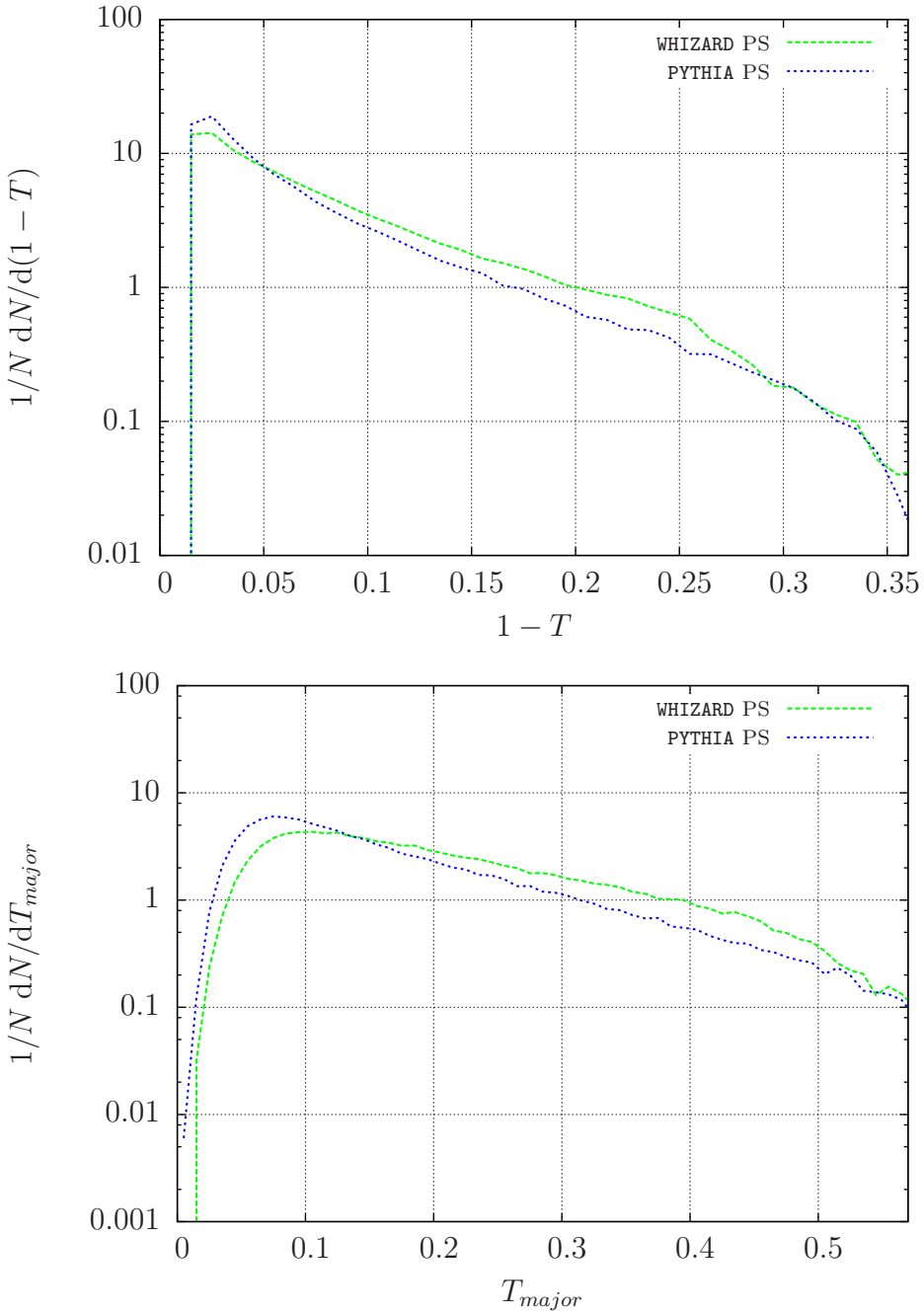


Figure 5.1: Plots for thrust $T$ and thrust major $T_{major}$ (without hadronization). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

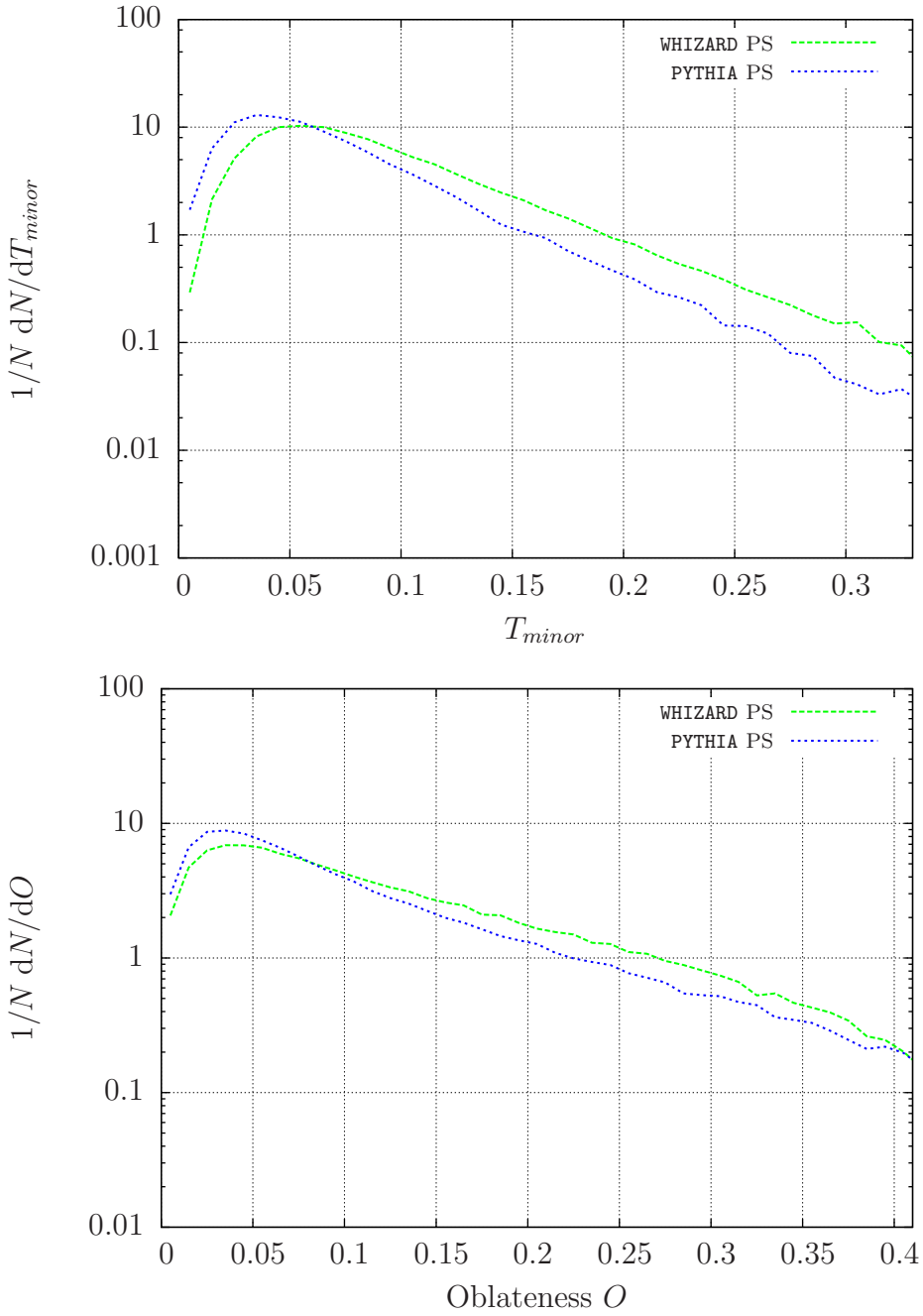


Figure 5.2: Plots for thrust minor $T_{minor}$ and Oblateness $O$ (without hadronization). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

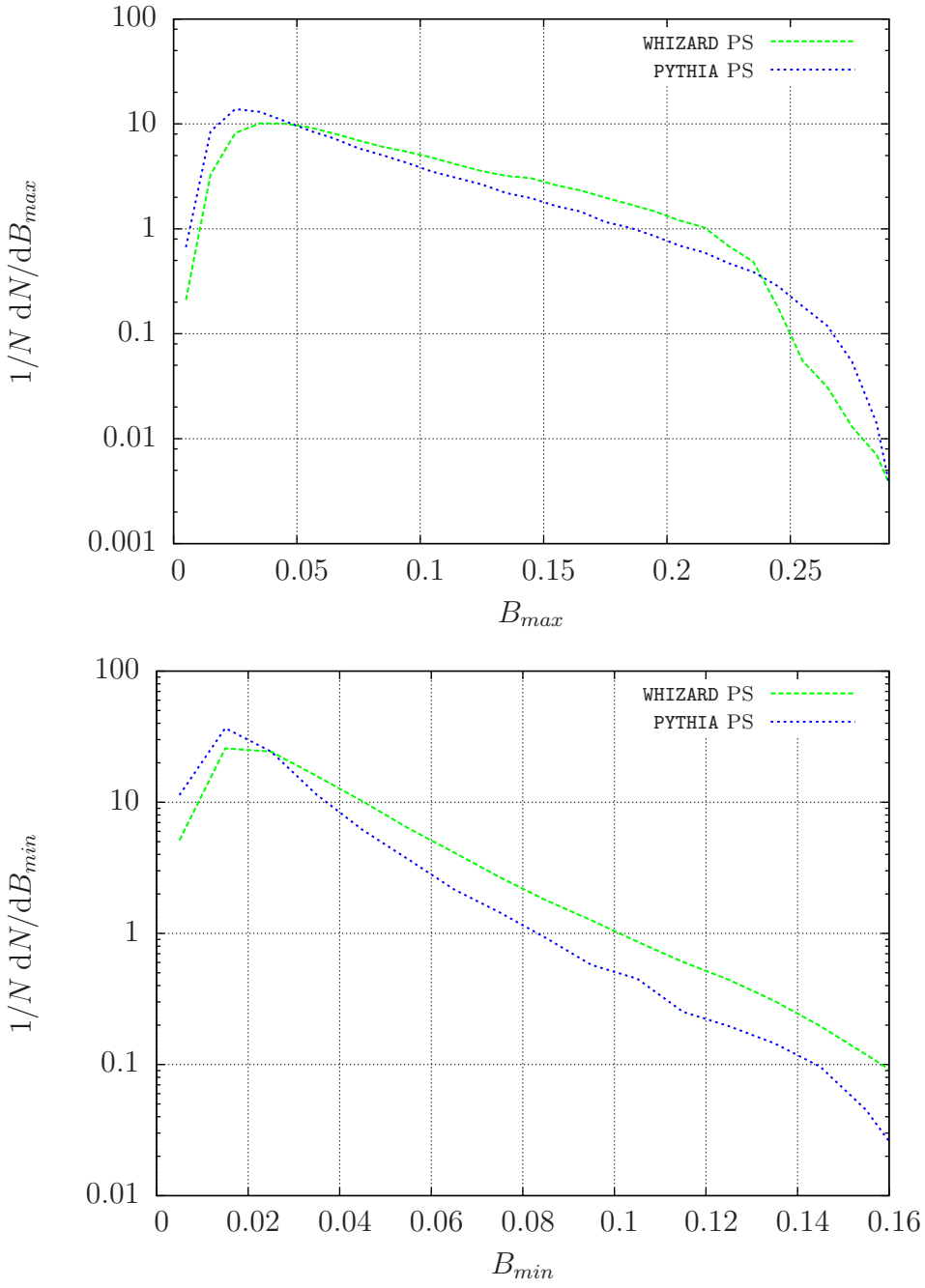


Figure 5.3: Plots for jet broadenings $B_{max}$ and $B_{min}$ (without hadronization). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

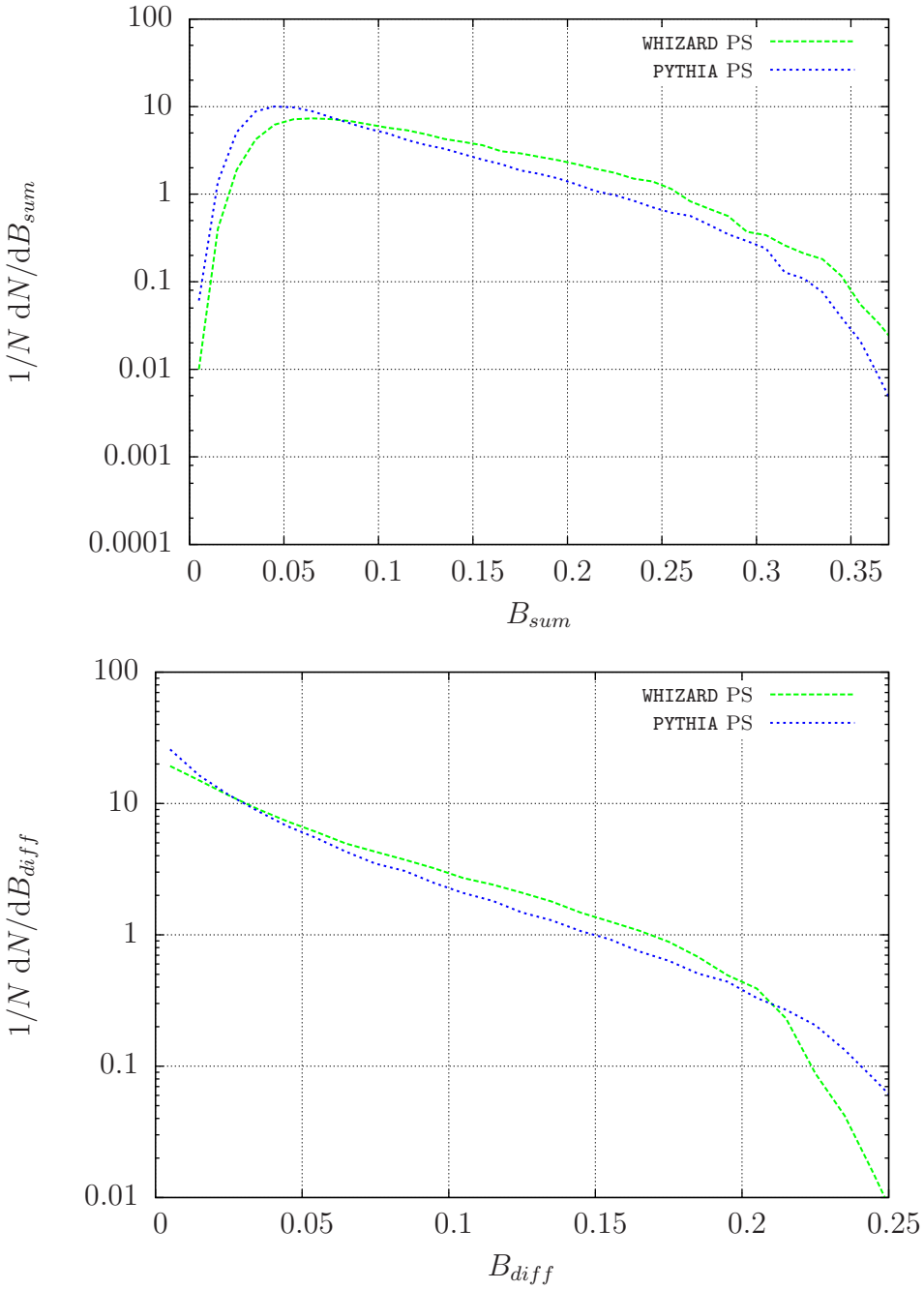


Figure 5.4: Plots for jet broadenings $B_{sum}$ and $B_{diff}$ (without hadronization). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

generates more spherical events compared to PYTHIA's parton shower. Nonetheless, they show a satisfactory agreement as WHIZARD's parton shower was not tuned at all for these plots. However, it is unclear if the discrepancies can be tuned away. Moreover, as distributions at parton level are not observable in an experiment, it is doubtful if they even need to be.

## 5.2 Final-State Radiation at Hadron Level

### 5.2.1 Event Shapes

For hadronized events, we can compare the generated distributions with experimental data. We compared the distributions for several event shapes with data from the DELPHI collaboration [87]. The measurement was performed using $e^+e^-$ collisions at center-of-mass energies of $\sqrt{s} = 130\,\text{GeV}$ and 136 GeV. The simulated hard interaction was chosen to be $e^+e^- \to u\bar{u}$ at a center-of-mass energy of $\sqrt{s} = 133\,\text{GeV}$.

The resulting distributions for both parton shower algorithms are shown in figures 5.5 to 5.8 for the event shapes already shown in the previous section and in figure 5.9 for the energy-energy correlation (EEC).

Both parton showers show good agreement, especially if one takes into account that the events showered with WHIZARD's parton shower where hadronized with the PYTHIA hadronization tuned to data using events showered with PYTHIA. As for the unhadronized samples, events showered with WHIZARD tend to populate the regions corresponding to more spherical configurations compared to events generated using the PYTHIA shower. The plot for thrust major $T_{maj}$ shows a slight undershooting of the WHIZARD curve with respect to the data in the two bins from 0.04 to 0.08. However, both distributions are mostly consistent with the data. The plot for the energy-energy correlation shows an overshoot in the $\cos\chi \to \pm 1$ bins and, accordingly, an overshoot in all other bins. As these bins correspond to soft emissions in events with back-to-back jets, it is likely to be an artifact of the hadronization.

### 5.2.2 Jet Rates

A comparison of the Monte Carlo results for the process $e^+e^- \to q\bar{q}$ at $\sqrt{s} = 91\,\text{GeV}$ with measurements from the JADE and OPAL collaborations given in [89] is shown in figures 5.10 and 5.11. Shown are differential jet rates as a function of the resolution parameter in the $k_T$-clustering algorithm $y_{i\,i+1}$, where the event turns from being a $i+1$-jet event into a $i$-jet event. The definition of the clustering variable is given in equation (A.6) in the appendix. The comparison is equivalent to the one in [90, 91], where a

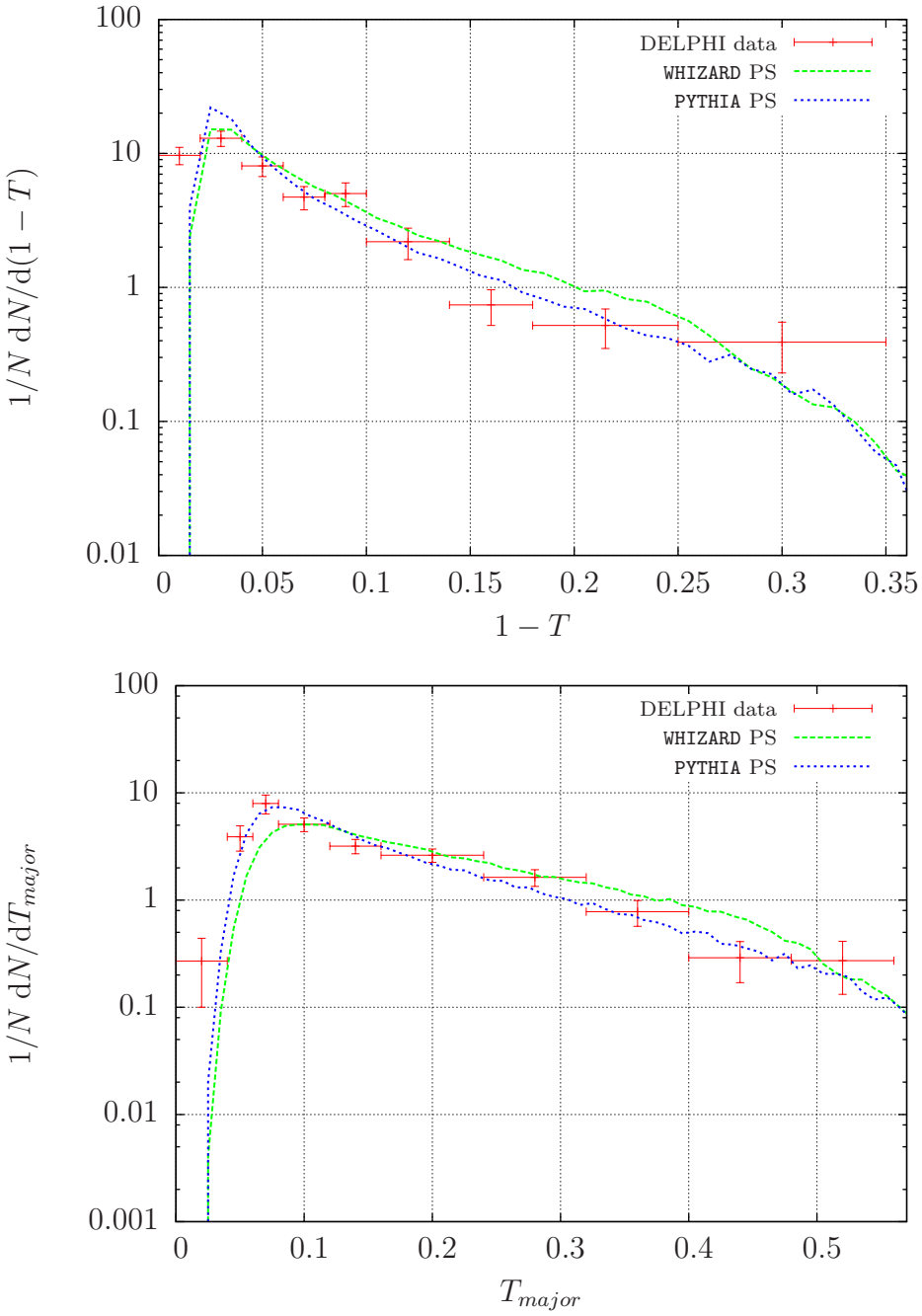


Figure 5.5: Plots for thrust $T$ and thrust major $T_{major}$ (with hadronization, data from [87]). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

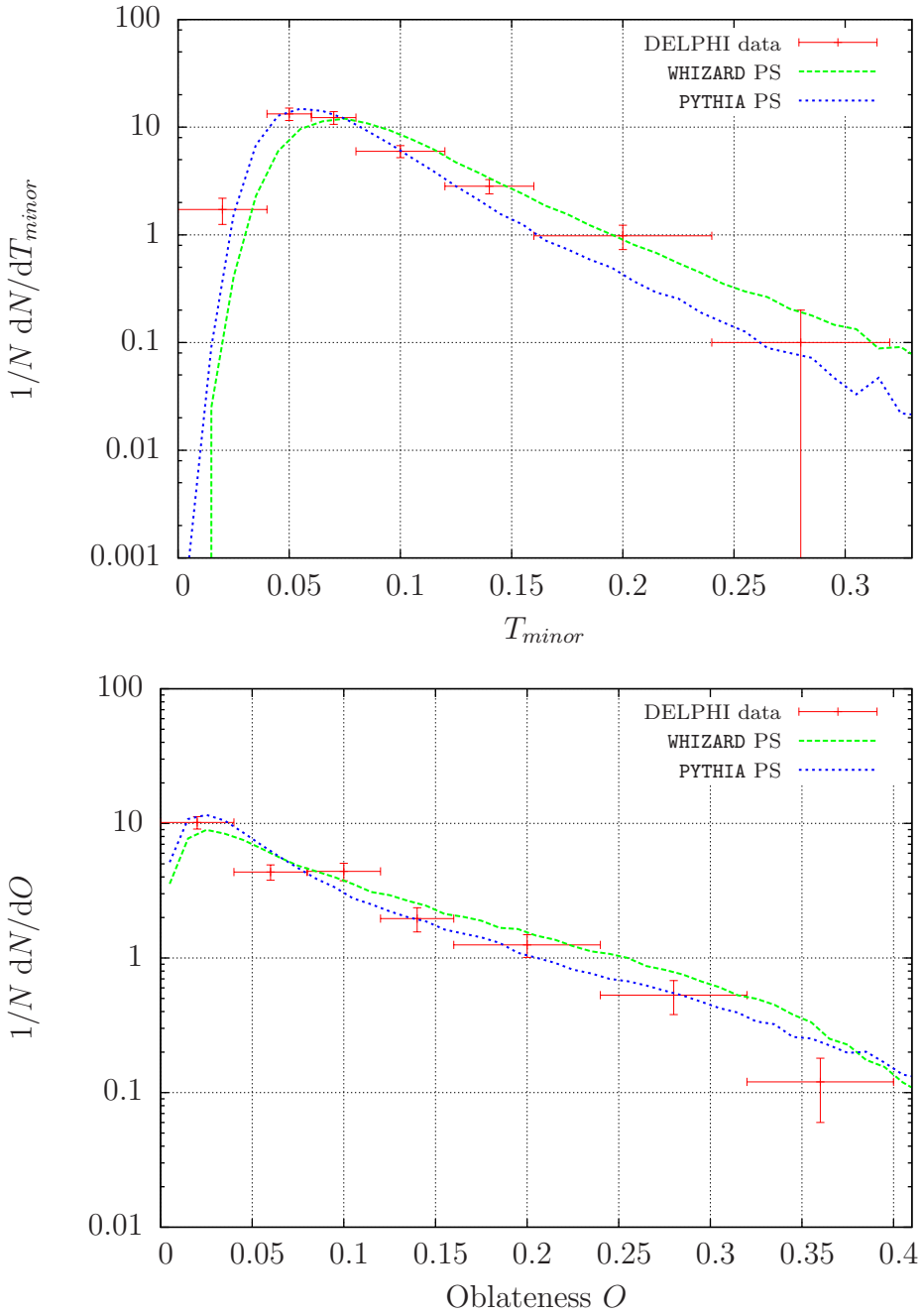


Figure 5.6: Plots for thrust minor $T_{minor}$ and Oblateness $O$ (with hadronization, data from [87]). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

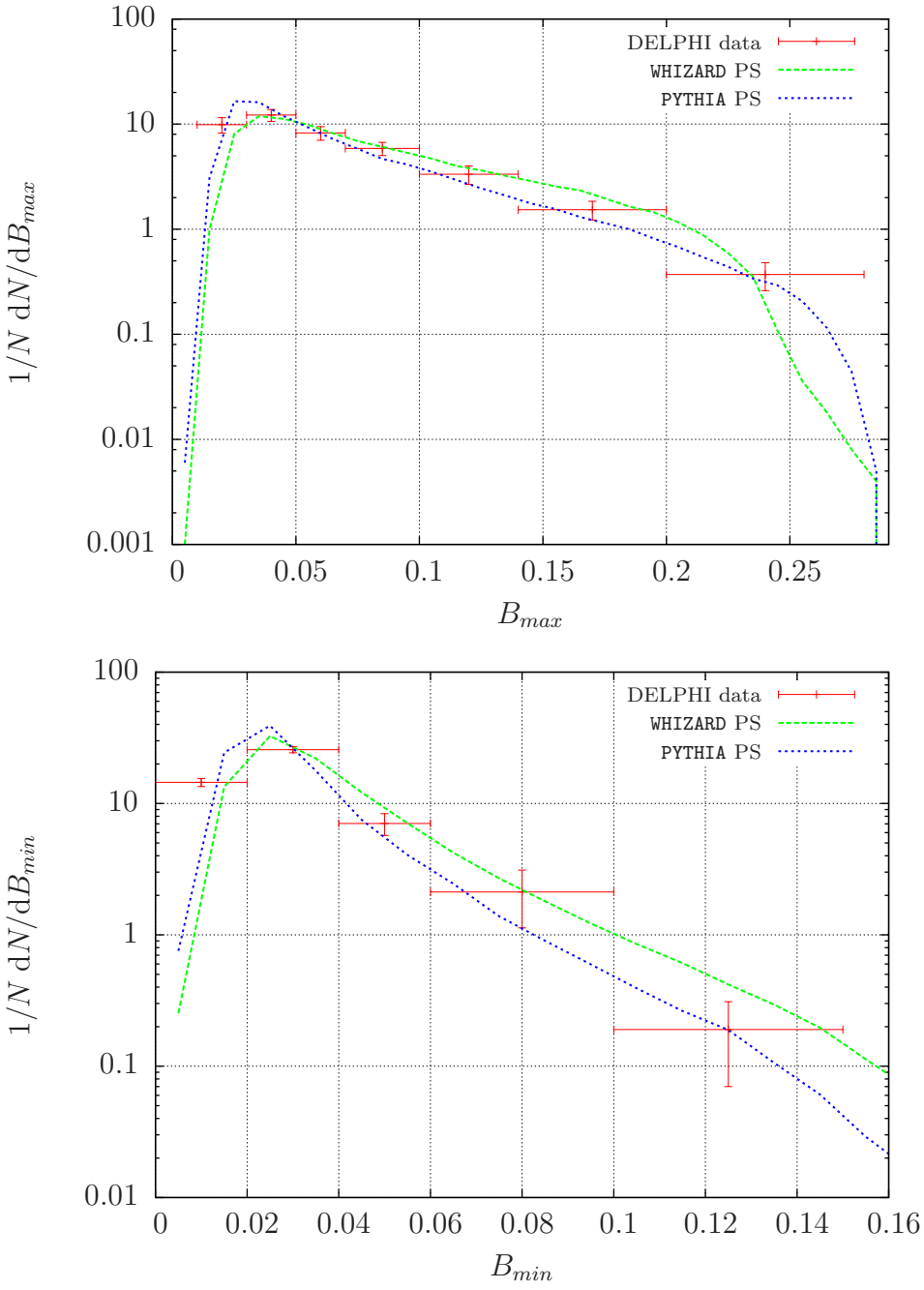


Figure 5.7: Plots for jet broadenings $B_{max}$ and $B_{min}$ (with hadronization, data from [87]). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

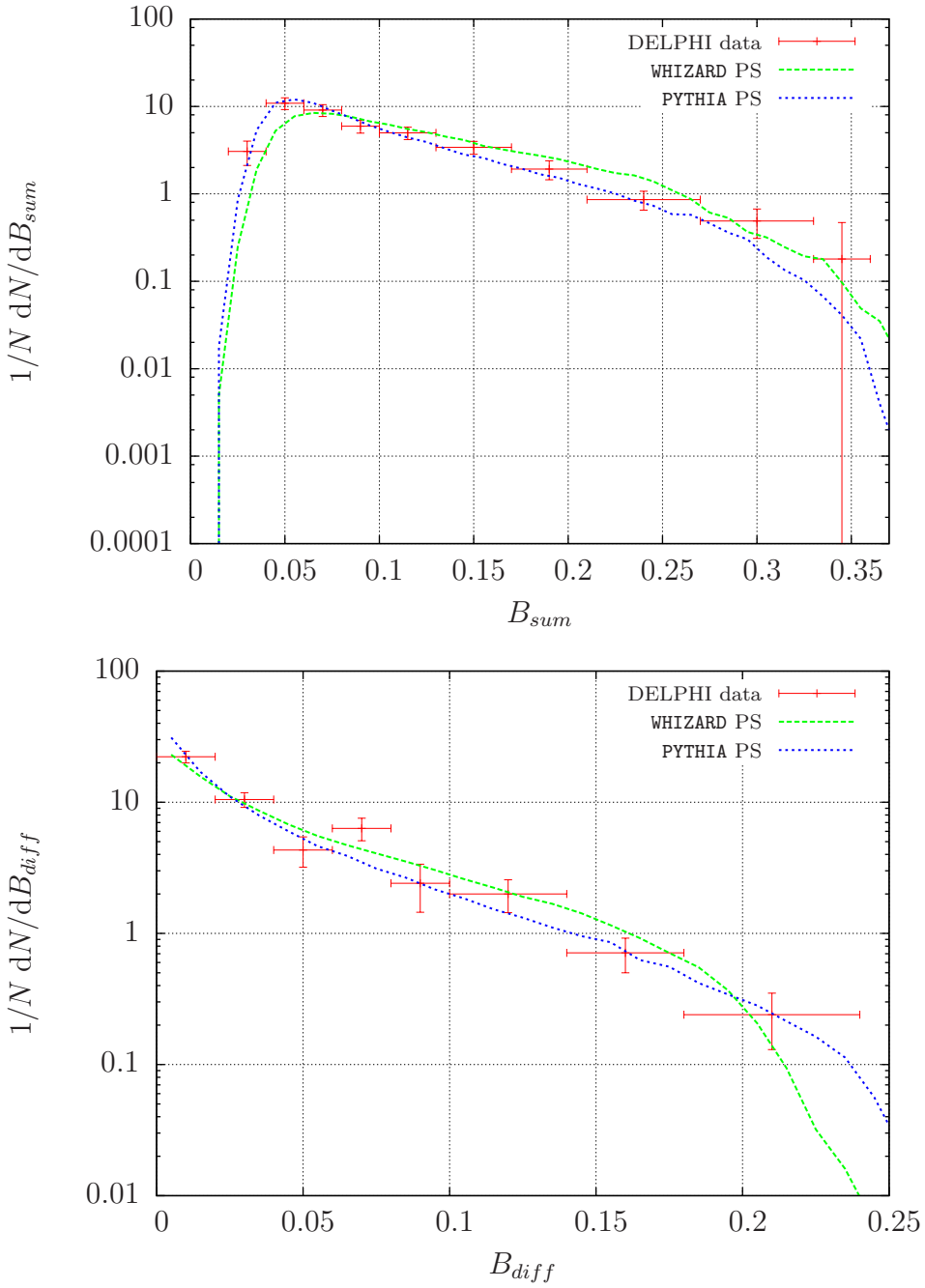


Figure 5.8: Plots for jet broadenings $B_{sum}$ and $B_{diff}$ (with hadronization, data from [87]). The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

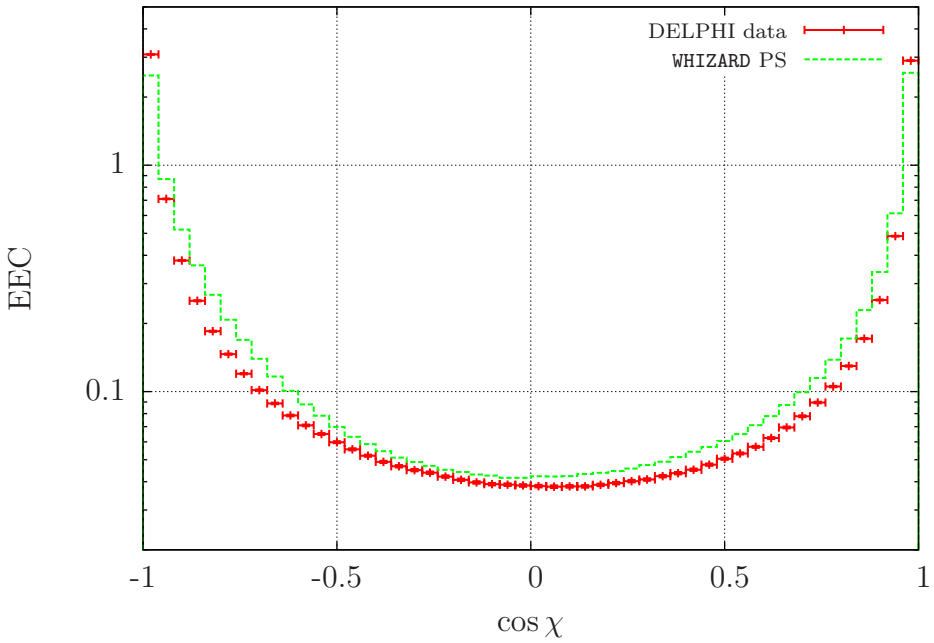


Figure 5.9: Plot for energy-energy correlation at LEP (with hadronization, data from [86], EEC calculated using Rivet [88]). The dashed/green/bright line is `WHIZARD`.

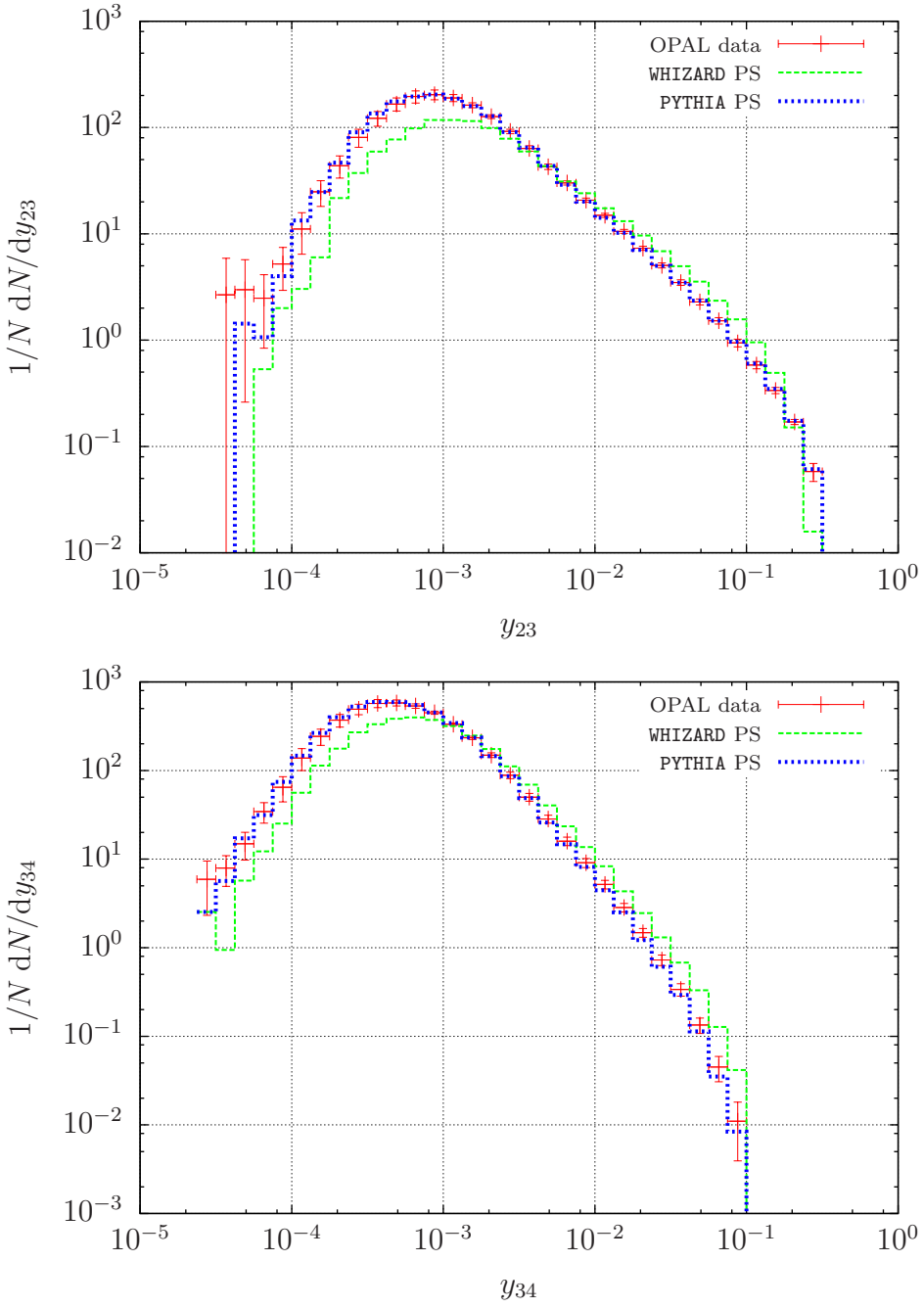


Figure 5.10: Plots for differential jet rates $y_{23}$ and $y_{34}$. The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

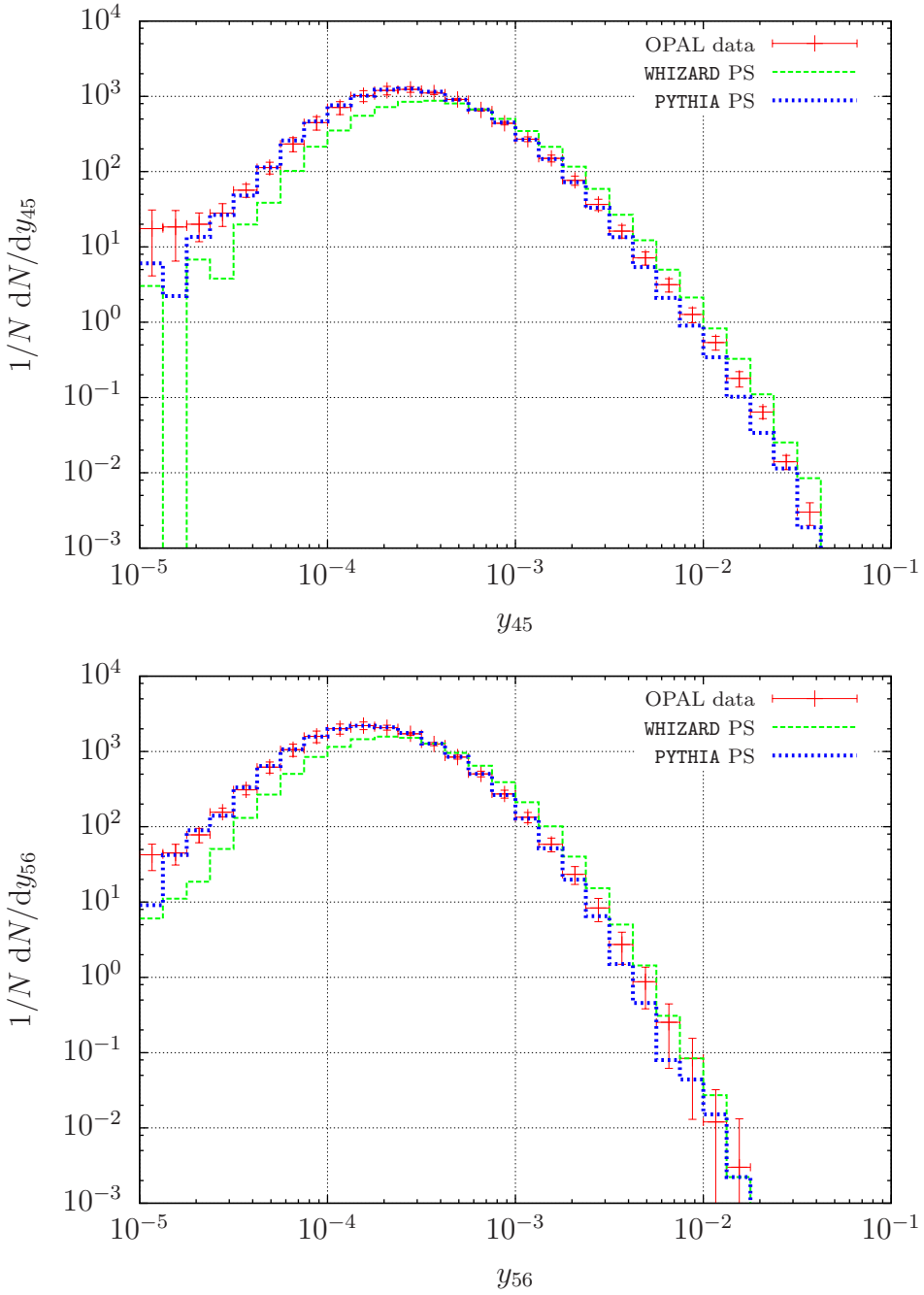


Figure 5.11: Plots for differential jet rates $y_{45}$ and $y_{56}$. The dashed/green/bright line is `WHIZARD`, the dotted/blue/dark line is `PYTHIA`.

tuning of some parton shower and hadronization parameters was performed. The only tuning applied to the parton shower in the comparison was a by-hand adjustment of $\alpha_S$, setting $\Lambda$ to a value of 0.15 GeV. In general, the plots confirm the tendency of WHIZARD's parton shower to generate more spherical events compared to PYTHIA as the bins with higher values of $y_{i\,i+1}$ are populated more. Note that small values of $y_{i\,i+1}$ correspond to small invariant masses and that these regions are described by the hadronization model and not the parton shower. So the differences in the left parts of the plots can stem from two sources. They might be caused by normalization effects due to over-estimation in the right parts. Any remaining difference would show that the hadronization tune obtained with PYTHIA is not suitable to describe these regions when used with WHIZARD's shower.

## 5.3 Initial-State Radiation

A plot for the transverse momentum of a $Z$-Boson produced in $p\bar{p}$-collisions at $\sqrt{s} = 1.96\,\text{TeV}$ is given in figure 5.12. The simulation with PYTHIA was done using Rick Field's CDF Tune D6 with CTEQ6L1 parton distribution functions. The simulation using WHIZARD's parton shower was done using the same PDFs, multiple interactions were disregarded in both simulations. The data obtained from WHIZARD's initial-state parton shower shows two distinct features: first of all, the curve in the low-$p_T$ region shows a slight deviation with respect to the corresponding PYTHIA curve. However, as we will see later, this is still in agreement with data. Second of all, it shows the known phase space cut at $p_T \lesssim m_Z$ [92]. For comparison, the plot is supplemented by a $p_T$-histogram for the unshowered process $u\bar{u} \to Zg$. PYTHIA's description uses the power-shower and matching and closely resembles the result for the partonic process.

Our approach to solve the shortcomings of WHIZARD's parton shower was not to include the power shower ansatz, but instead accept this as a deficiency of the parton shower and delegate the task of describing the high-$p_T$ region to a matching algorithm.

## 5.4 Matched Final-State Radiation

Plots for results obtained with the MLM matching for the final-state parton shower are shown in figures 5.13 and 5.14 for events showered with WHIZARD's parton shower and figures 5.15 and 5.16 for PYTHIA. The process under consideration is $e^+e^- \to u\bar{u}$ at a center of mass energy of 91 GeV, hadronization was switched off. The process was simulated in five different ways, first without any matching at all and then with a variable number of additional jets

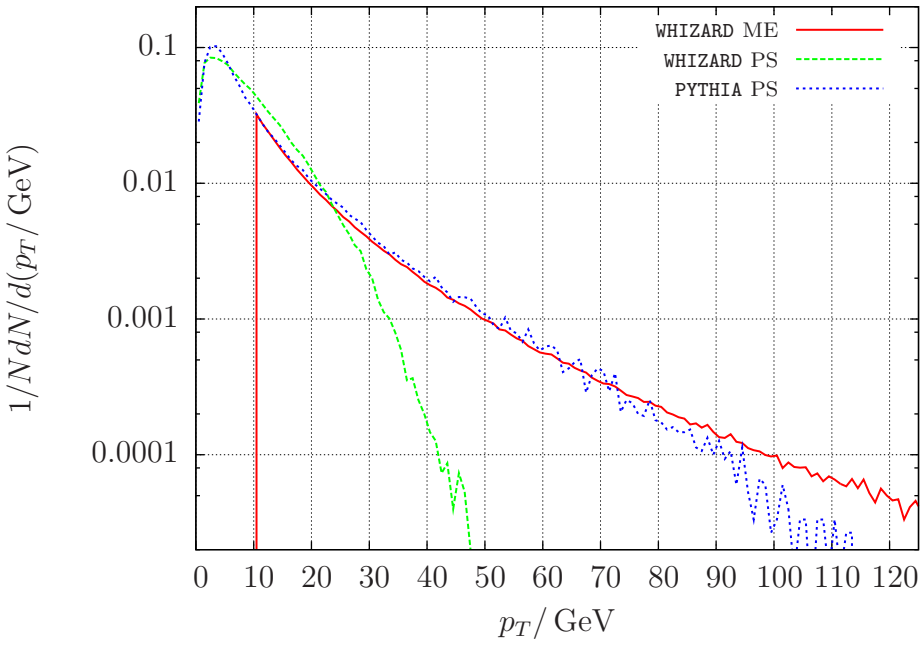


Figure 5.12: Transverse momentum of a $Z$-Boson in various schemes. The normalization for events from WHIZARD's matrix element was chosen manually to fit PYTHIA's PS result in the range $10\,\mathrm{GeV} < p_T < 20\,\mathrm{GeV}$.

from zero to three, where each additional jet could be a gluon or a $u$,$d$,$s$ or $c$ quark. For the unmatched case and for each jet multiplicity an event set consisting of 150000 events was simulated. The plots show normalized distributions for event shapes obtained from these samples.

The plots show some common features. The line for the (moot) case of no additional jets closely resembles the line for the unmatched event sample, except in the region of low thrust (right part of the upper image in figures 5.13 and 5.14). The missing events are events where the parton shower splits a hard jet into two separated jets, so that the matching procedure cannot cluster any of the two jets to the original parton and therefore rejects the event.

The lines for one, two and three additional jets lie on top of each other so that it can be concluded that for these observables, the inclusion of one additional jet is sufficient. The deviations between the unmatched and the matched event samples exhibit different behavior: for `PYTHIA` the number of spherical events is larger for the matched sample, stemming from the better description of large angle emissions. For `WHIZARD` the deviations are opposite, the number of more pencil-like events are enhanced, while especially the number of events with medium values of $1 - T$ and $T_{maj}$ is decreased. This can be regarded as correcting the tendency to favor more spherical events mentioned in section 5.2. The differences between the distributions for matched and unmatched event samples have to be taken into account when tuning the combination of shower and matching to data. Therefore this can be seen as an example for an observable which is sensitive to regions enriched by hard jet emission, and not so much dominated by universal logarithmic terms. For such an observable, a tuning obtained without matching cannot be reliably used to generate matched samples.

We also did a comparison to data corresponding to the comparison for the unmatched showers in figure 5.5. We used the curve for the $e^+e^- \to 5jets$ as the sample for the matched shower. The plot is shown in figure 5.17. The curve for thrust $T$ is slightly altered, most prominent differences to figure 5.5 is a less pronounced peak with both showers and an increase for the `PYTHIA` curve for values $1-T > 0.1$. The curves for Thrust major $T_{maj}$ show similar behavior to the unmatched curves. Both reproduce the data, except for `WHIZARD`'s parton shower's tendency to more spherical configurations and the small number of events in the lower $T_{maj}$-bins. Both these deficiencies have already been visible in the unmatched event samples.

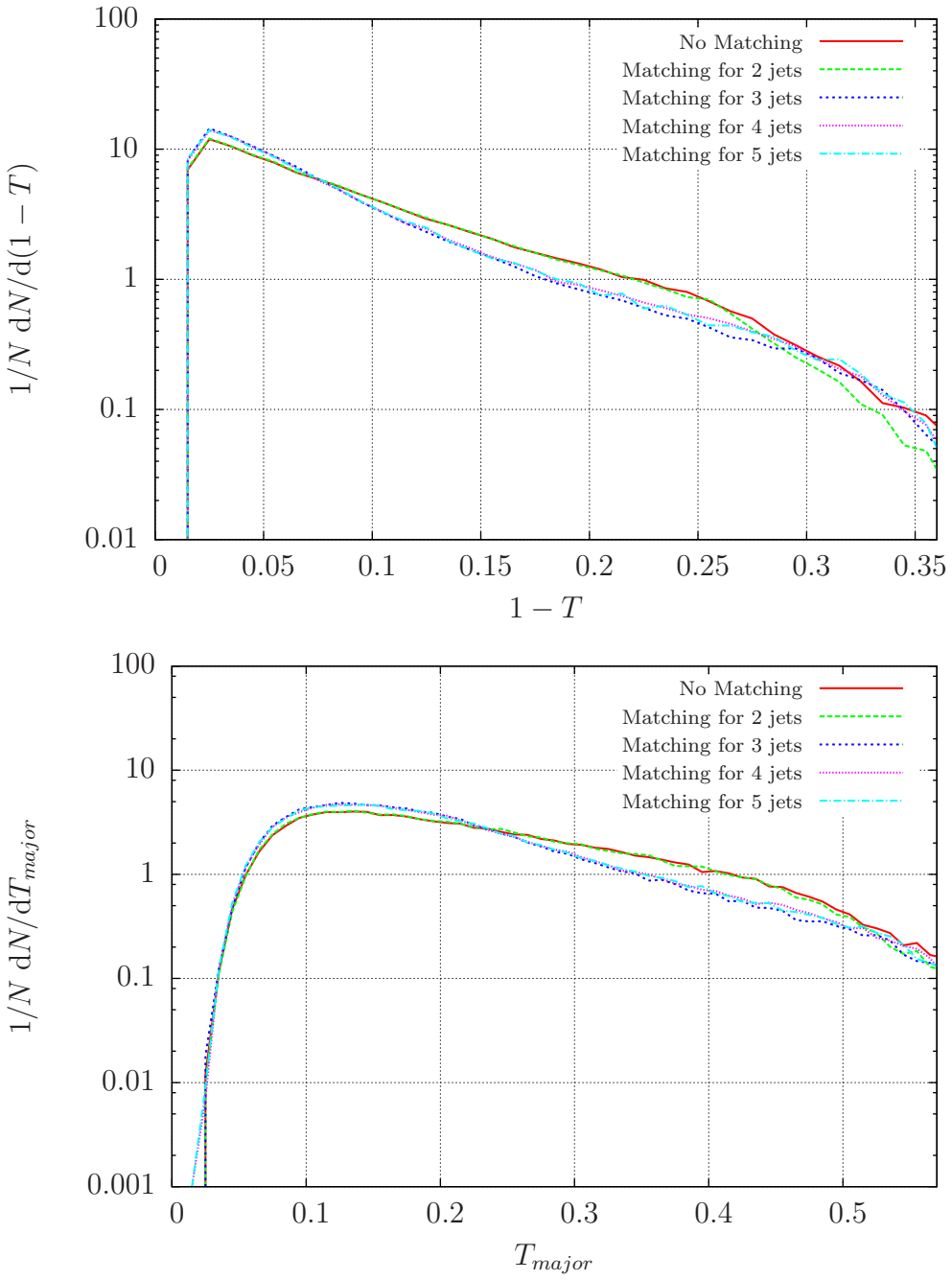


Figure 5.13: Plots for thrust $T$ and thrust major $T_{major}$ (`WHIZARD` ME + `WHIZARD` PS with matching).

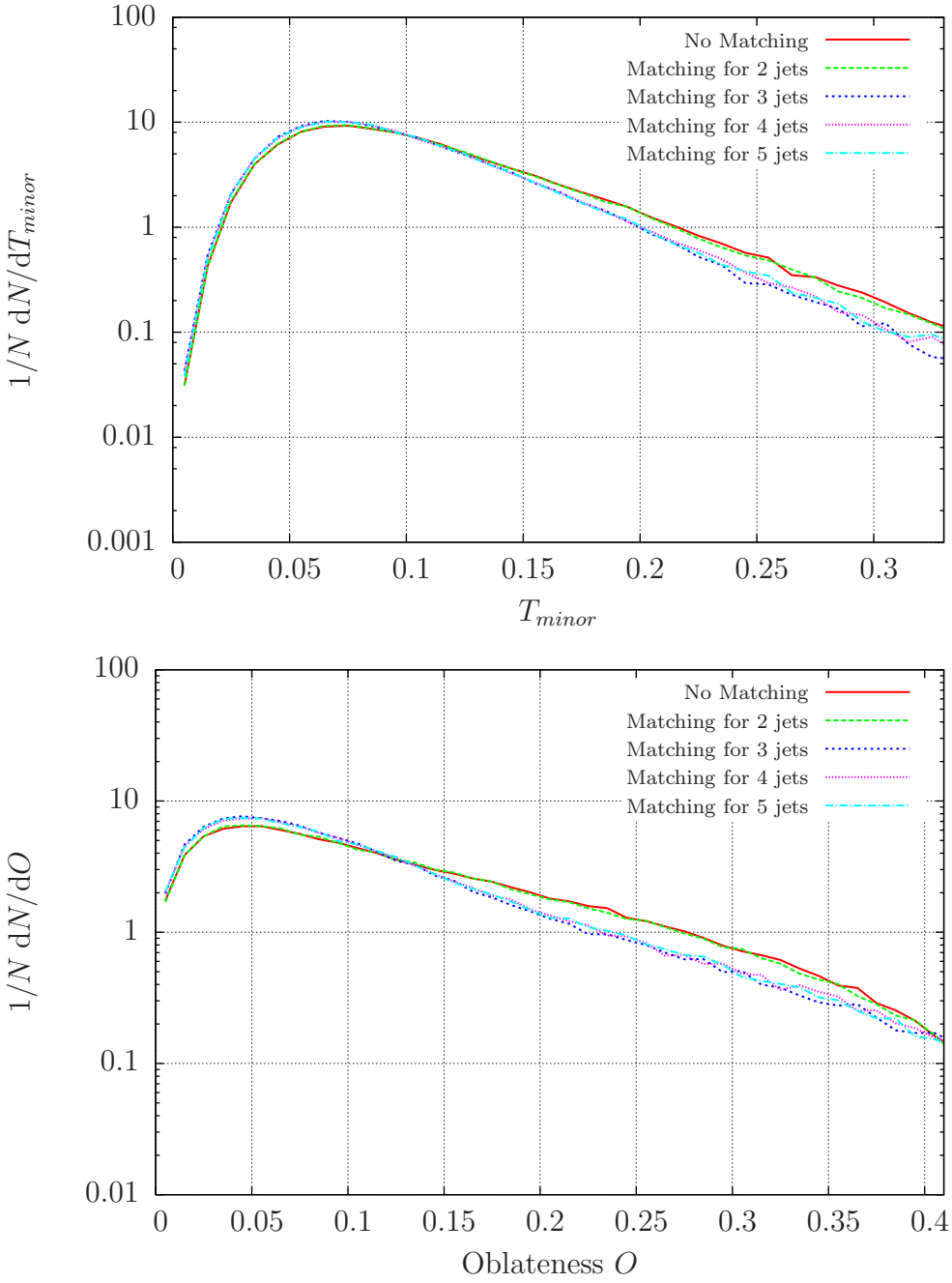


Figure 5.14: Plots for thrust minor $T_{minor}$ and Oblateness $O$ (`WHIZARD` ME + `WHIZARD` PS with matching).

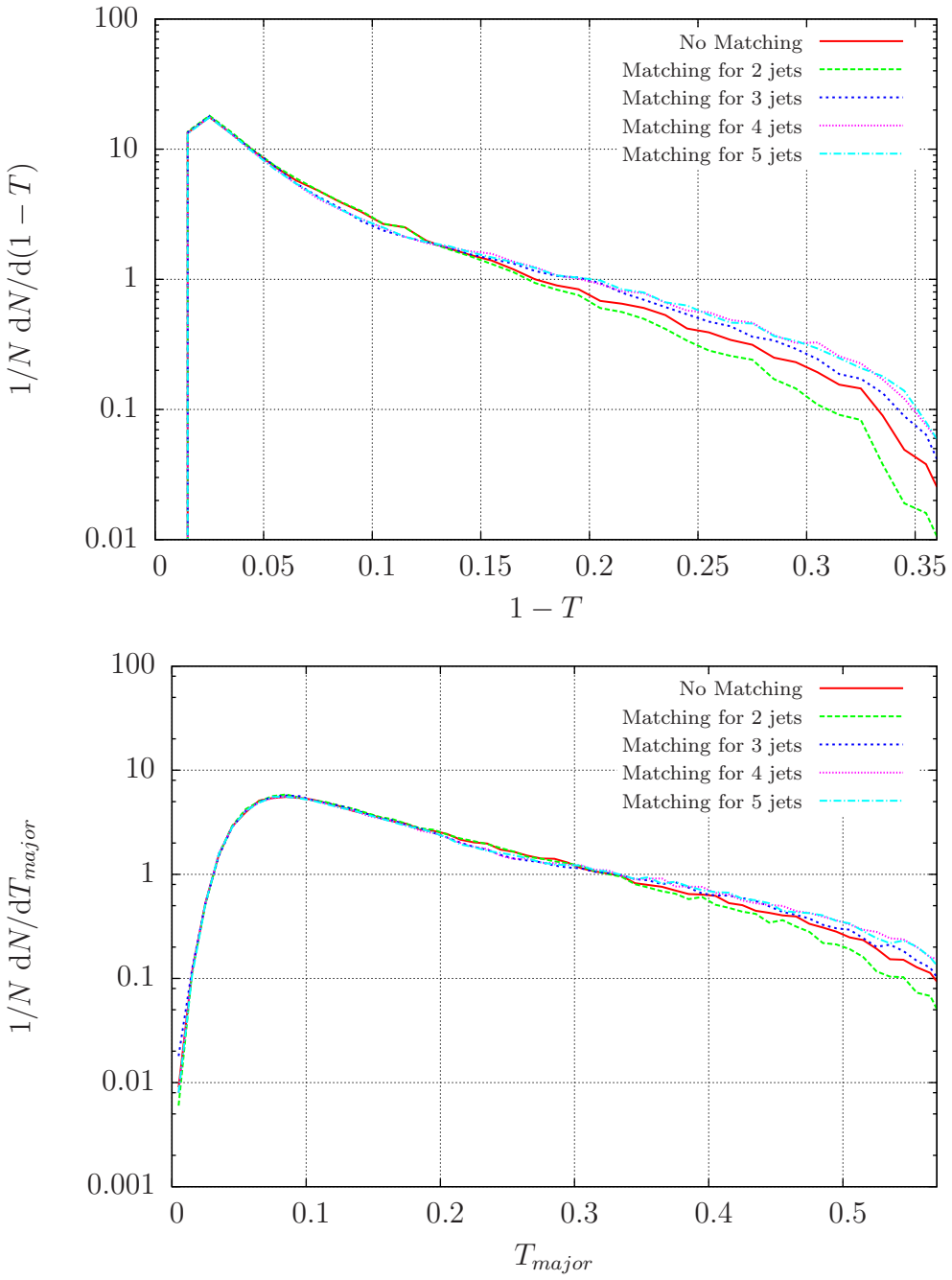


Figure 5.15: Plots for thrust $T$ and thrust major $T_{major}$ (`WHIZARD` ME + `PYTHIA` PS with matching).

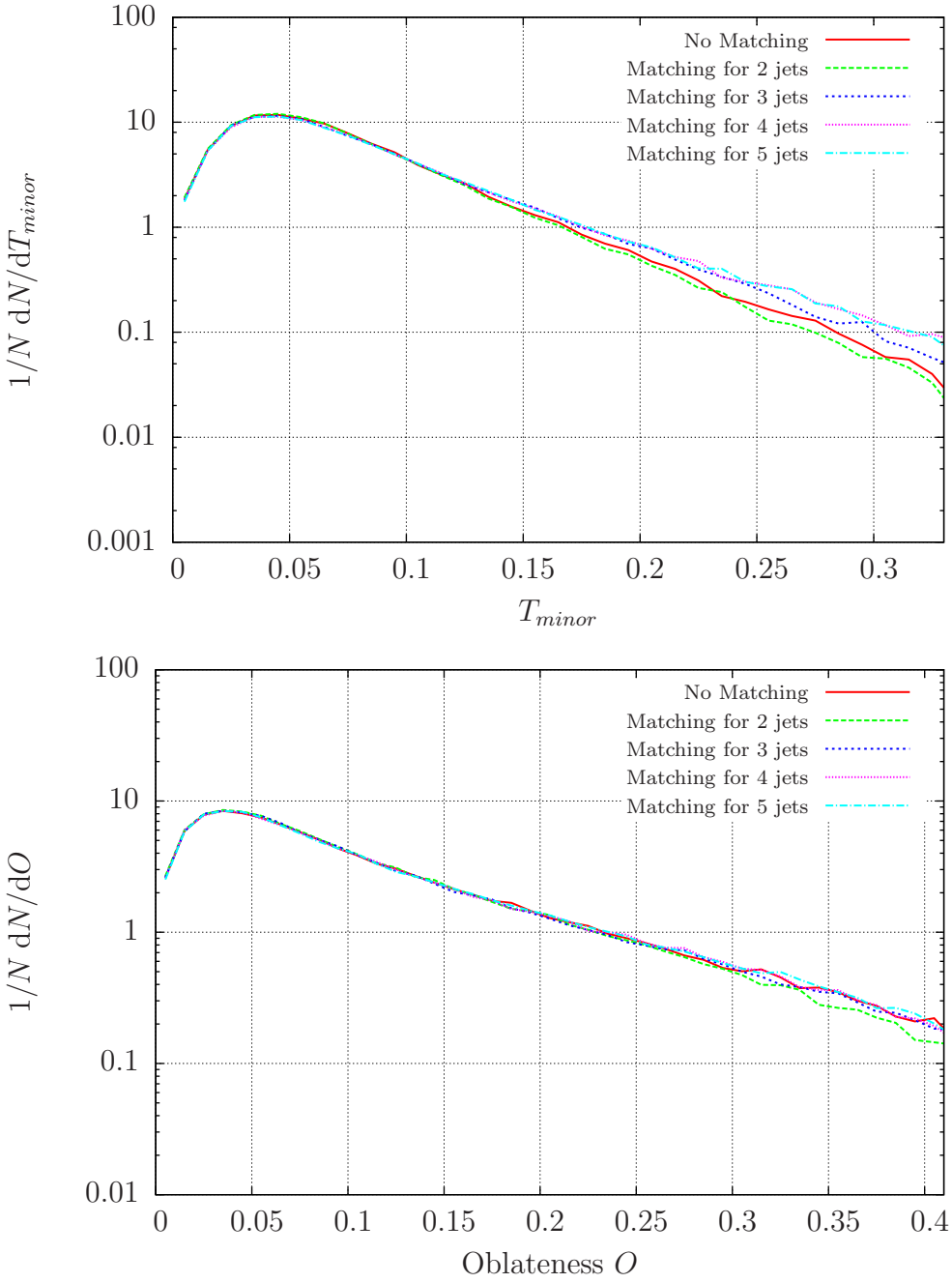


Figure 5.16: Plots for thrust minor $T_{minor}$ and Oblateness $O$ (`WHIZARD` ME + `PYTHIA` PS with matching).

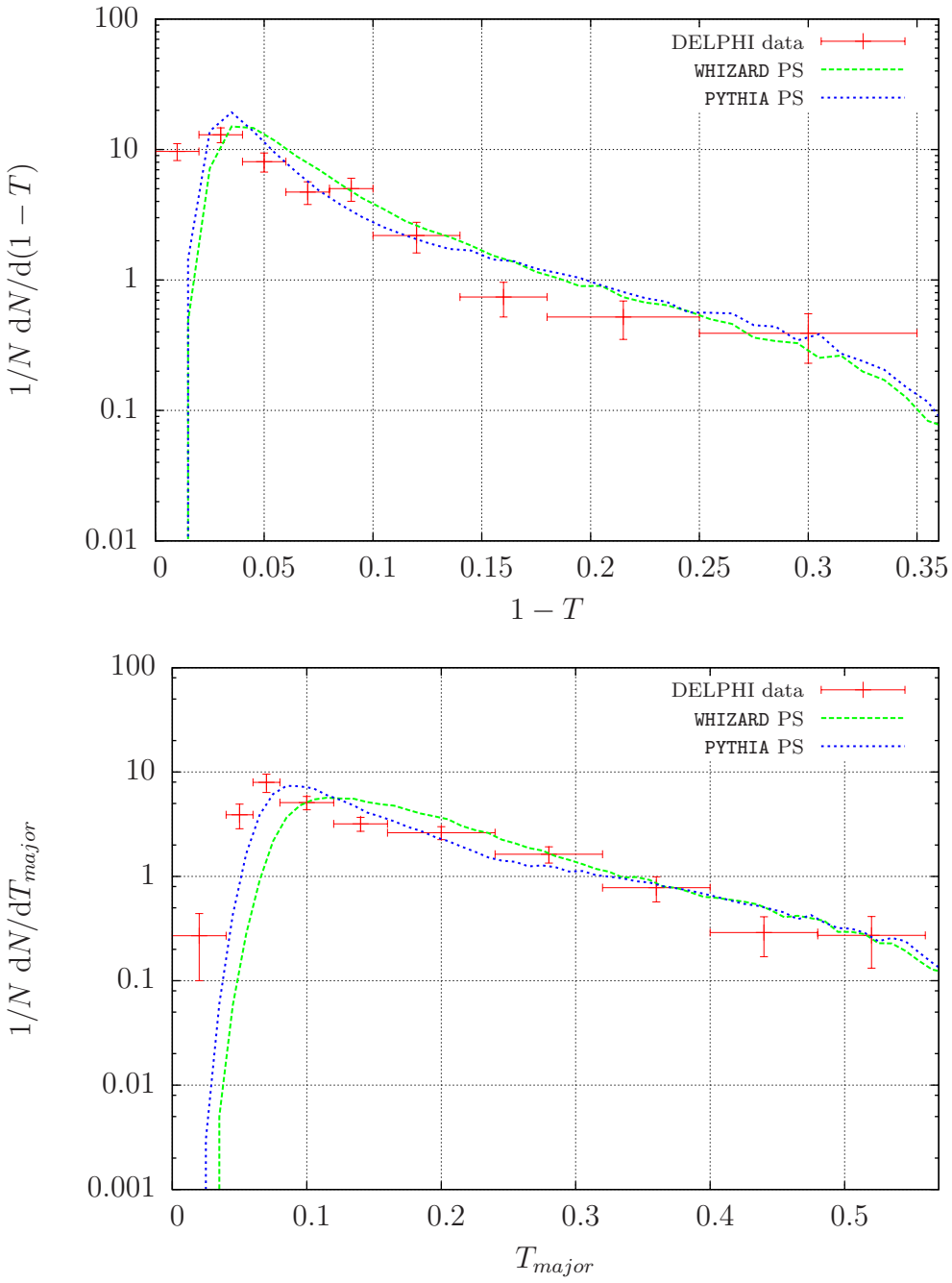


Figure 5.17: Comparison of the predictions for parton showers matched to the process $e^+e^- \to u\bar{u} + 3jets$.

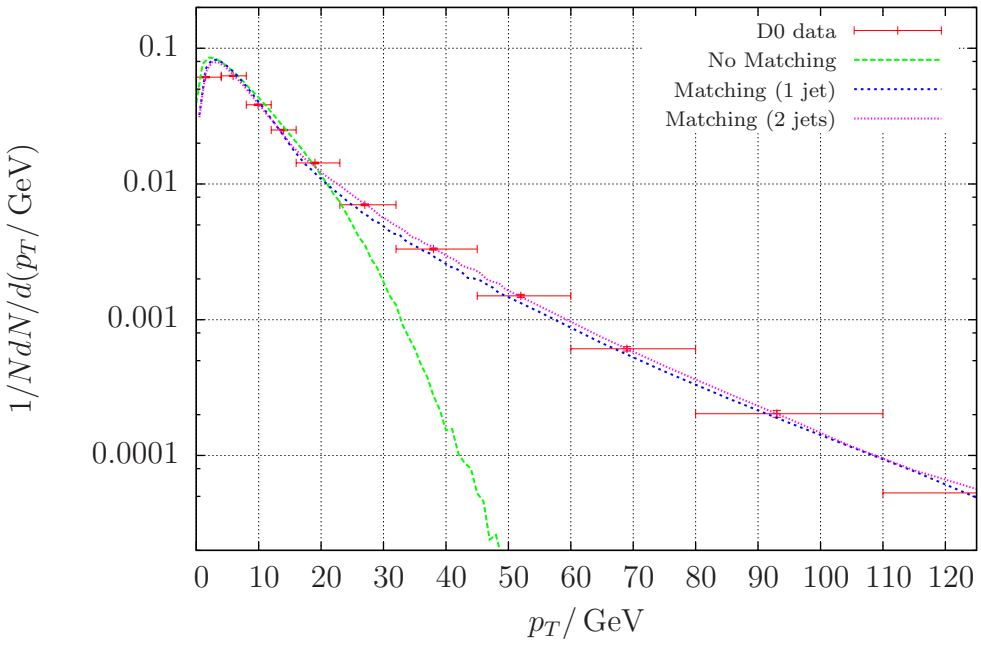


Figure 5.18: $Z$-Boson transverse momentum, simulated with `WHIZARD` ME and PS without and with matching.

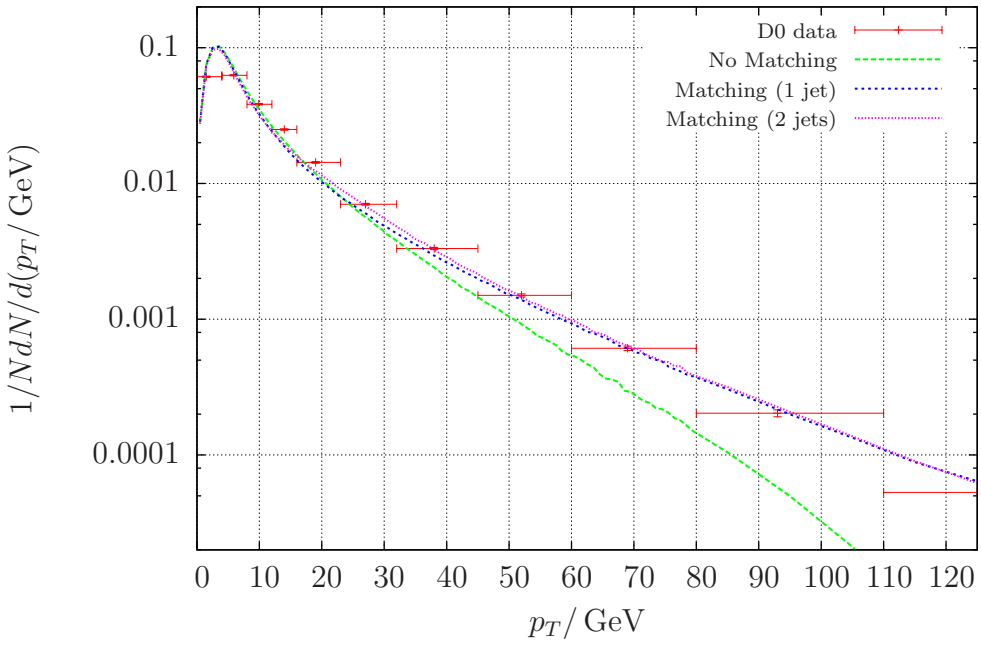


Figure 5.19: $Z$-Boson transverse momentum, simulated with `WHIZARD` ME and `PYTHIA` PS without and with matching for one and two additional jets.

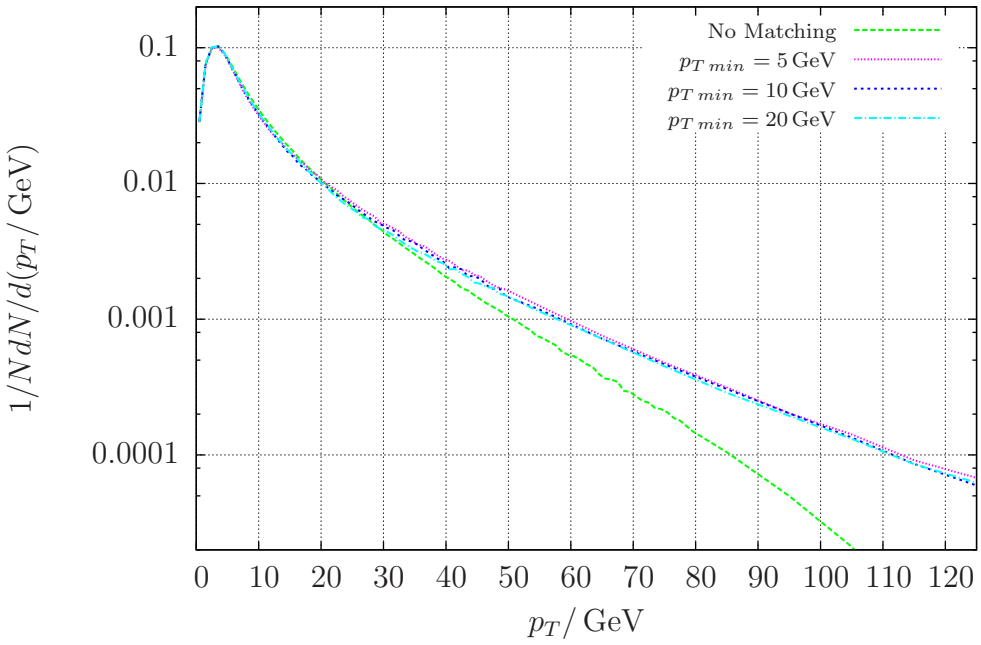


Figure 5.20: The "NoMatching" curve and the "Matching (1jet)"(cf. figure 5.19) curve for three different values of $p_{T\,min}$ (5 GeV, 10 GeV and 20 GeV).

## 5.5 Matched Initial-State Radiation

To test the matching procedure for the initial state we simulated the process $q\bar{q} \to Z$ and the additional corrections $j\,j \to Z\,j\,(j)$ for one (two) additional jets, $j = u, \bar{u}, d, \bar{d}, s, \bar{d}, c, \bar{c}, g$. The resulting distributions for the $Z$ boson transverse momentum are given in figure 5.18 for WHIZARD's parton shower and figure 5.19 for PYTHIA's parton shower. For comparison the measured distribution from D0 [93] was included. Note that all simulated distributions were obtained with disabled primordial $k_T$.

As expected, the results for PYTHIA do not depend much on the application of matching as its power shower approach already generates a $p_T$-distribution close to the correct distribution [92][31]. The plot for WHIZARD shows the expected addition of high-$p_T$ events, the concavity is weakend. Adding a second jet described by the matrix element does change the distribution only marginally for both showers.

Figure 5.20 shows the dependence of the $p_T$-spectrum on the MLM matching parameter $p_{T\,min}$. The distribution should be independent of $p_{T\,min}$, however a small difference is visible in the range $10\,\text{GeV} \lesssim p_T \lesssim 80\,\text{GeV}$. The high-$p_T$-tail remains stable when changing $p_{T\,min}$, the shape at the peak does not change as well. The differences are within the expected dependence on the matching parameters.

As a further test, we compared the $Z$-Boson $p_T$ at the LHC. We used the recently published measurement by CMS [94]. Except for the change from proton-antiproton beams to proton-proton beams and the increased center of mass energy $\sqrt{s} = 7\,\text{TeV}$, all other settings were the same as for the Tevatron simulation. This holds particularly for the chosen PYTHIA tune, that was obtained from measurements at Tevatron and usage at the LHC cannot be regarded as trustworthy. Nevertheless, the data can be reproduced very well, except for an overshoot in the lowest bins. As for WHIZARD there are no available tunes yet, so the dependency on a particular tune is not an issue. Note that the simulation was done with primordial $k_T$ disabled, so that the lowest bins are expected to be overpopulated. Apart from this difference, the simulation using WHIZARD's hard interaction, parton shower and matching procedure reproduces the data as good as the simulation performed using WHIZARD's hard interaction and matching, but PYTHIA's parton shower.

## 5.6 One Reweighting Example

As a first step towards reweighting events showered with an analytic shower, we implemented a reweighting procedure assuming two different constant

[31] PYTHIA's own matching was disabled during this simulation.

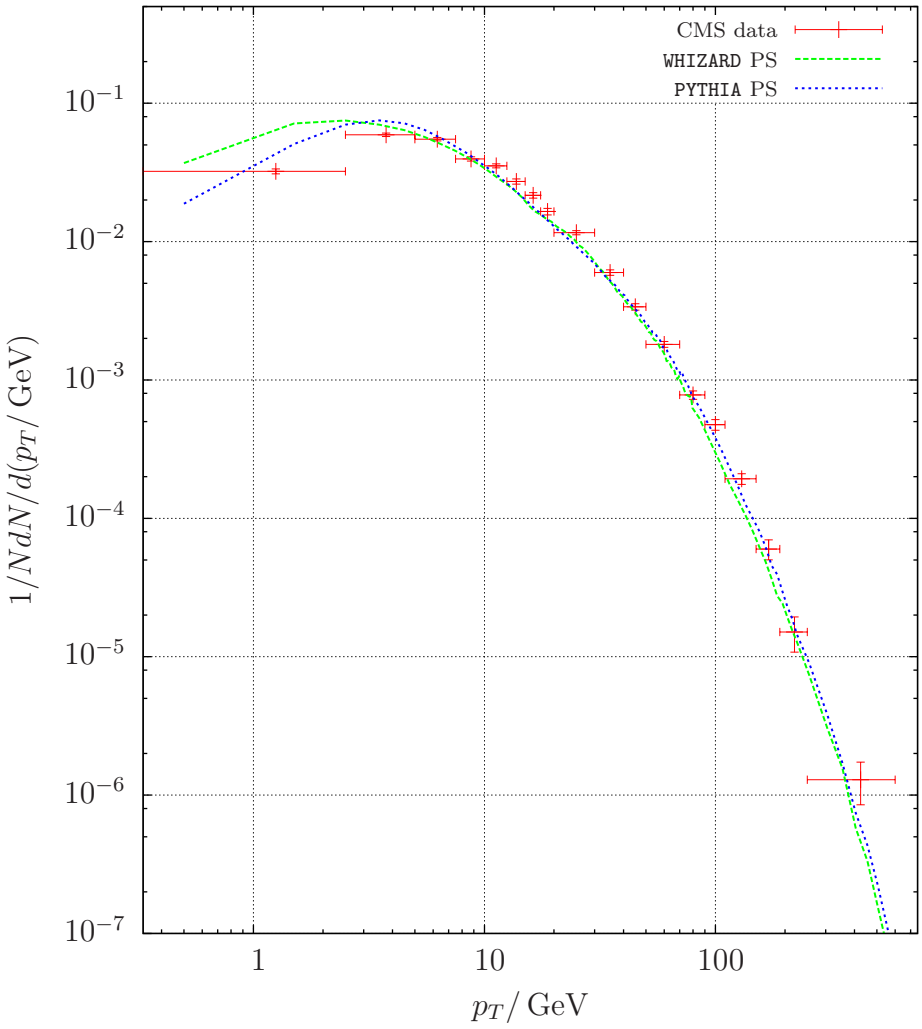


Figure 5.21: $Z$-Boson transverse momentum, simulated with WHIZARD ME and PS and WHIZARD ME + PYTHIA PS. A similar figure with the same data appeared in [94]. The dashed/green/bright line is WHIZARD, the dotted/blue/dark line is PYTHIA.

values for $\alpha_S$, 0.2 and 0.3. For these values, event samples for the processes $e^+e^- \to q\bar{q}$ with subsequent showers and a center-of-mass energy of $\sqrt{s} = 91\,\text{GeV}$ were generated, the resulting distributions are shown in figure 5.22. Then the events obtained using $\alpha_S = 0.3$ were reweighted to the respective event weights obtained using the setting $\alpha_S = 0.2$. The distribution of the reweighted events along with the actual $\alpha_S = 0.2$ distribution are shown in figure 5.23. The reweighted distribution resembles the original one very well, apart from a slight overshot in the highest bin and an underestimation in the central region. As the plot shows a normalized distribution, these two effects are probably connected. We did not investigate wether these discrepancies stem from statistical fluctuations or from systematic errors.

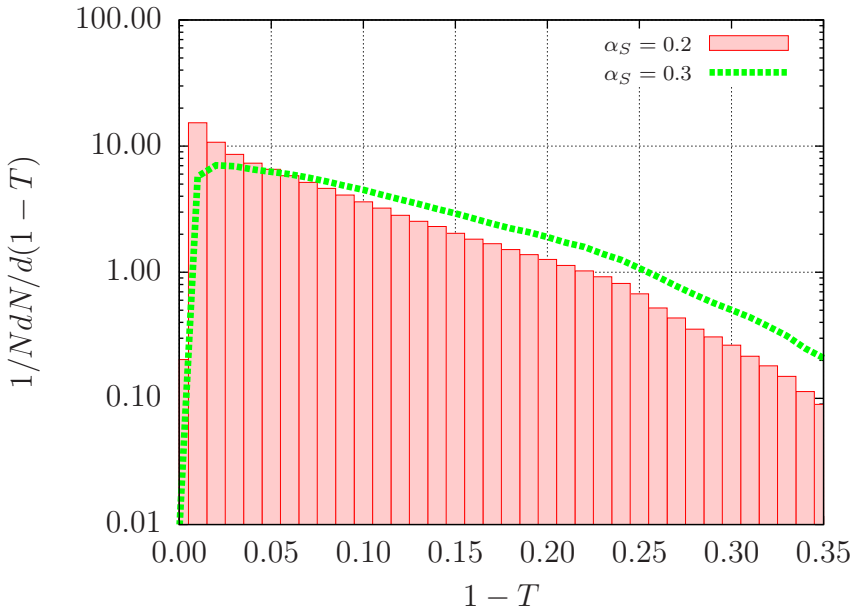


Figure 5.22: Thrust distribution, simulated using constant values of $\alpha_S = 0.2$ (red boxes) and $\alpha_S = 0.3$ (green curve).

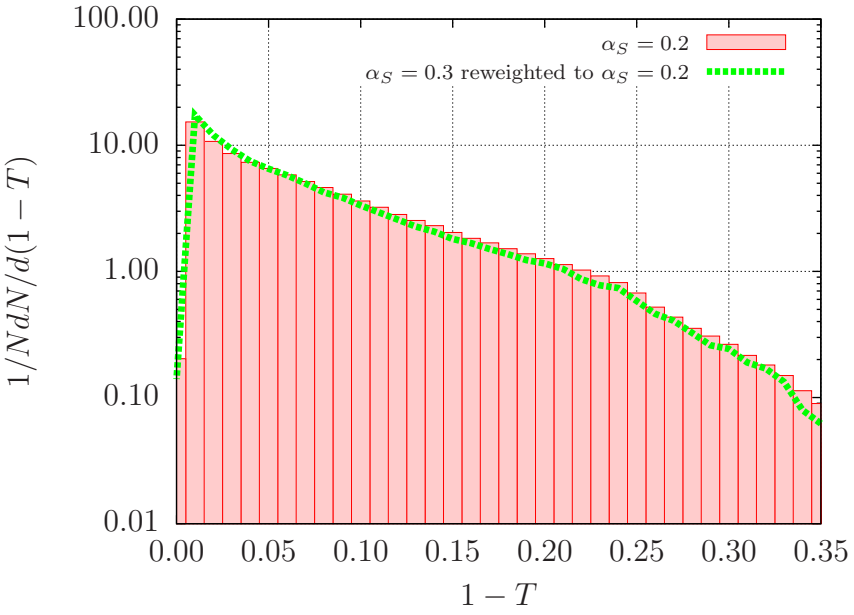


Figure 5.23: Thrust distribution, the data simulated using a $\alpha_S = 0.3$ was reweighted to $\alpha_S = 0.2$ (green curve), the distribution corresponding to $\alpha_S = 0.2$ (red boxes) remained unchanged.

# 6 Conclusions and Outlook

The recent decades have seen continuous experimental progress in the field of elementary particle physics. This progress has accompanied the progress in the theoretical description, that, among others, led to today's event generators. These event generators are highly-integrated software packages that include various subprograms. Each subprogram implements a particular aspect of the physics under consideration. Combining these different aspects and therefore also combining different programs is a task necessary for the understanding of today's event generators, which is in turn necessary for the understanding of today's experiments in elementary particle physics.

In this thesis, we presented an algorithm for an analytic parton shower for both initial- and final-state radiation. While a simplified version for final-state radiation has been known for quite some time, a complete treatment of final-state radiation and an extension to the initial state had not been available up to now.

Analytic parton showers are especially interesting for conceptual development in a theoretical description of QCD in a hadron collider environment (but also processes with hadronic final states at lepton colliders), as they allow to determine the corresponding shower weights from the complete shower histories. The knowledge of complete shower histories and weights enables one to e.g. change the hard scattering matrix element or the PDFs and reweight the showered events to the new hard scattering process. Furthermore, analytic parton showers might offer the possibility to determine systematic uncertainties from a parton shower approximation in a reliable and theoretically well-defined way. Also, it might be achievable – using analytic parton showers – to systematically construct higher-order corrections to the parton shower approximation.

One technical point that was found to be of particular importance for a successful algorithm for an analytic initial-state parton shower is the scale choice, specifically the starting condition of the backward shower evolution, together with the prescriptions for energy and momentum projections in the splittings. For the final-state shower, we also improved on the original algorithm, where e.g. running couplings constants within the shower evolution had not been taken into account.

For the description of complete kinematical distributions at hadron colliders, including the high-energy tails, we refrained from the power-shower concept, where also hard and/or non-collinear jets are being produced by means of the shower. Instead we use an MLM-type matching of the analytic parton shower with matrix elements containing one or more additional hard jets explicitly.

In this thesis, we presented our implementation of the analytic parton shower algorithm and our implementation of the MLM matching scheme and their integration into the event generator WHIZARD. Additionally we described the various interfaces needed to integrate PYTHIA's parton shower and hadronization routines. Thus all ingredients needed to simulate realistic events are available from within the framework of the WHIZARD generator. This setup also allows for a direct comparison of the PYTHIA and our own parton shower using the same hard matrix elements. During the implementation, we performed much technical work that entered into the development of the core WHIZARD program and cannot be mentioned here.

Together with the development of the algorithm, we made a thorough comparison with the results obtained using PYTHIA's parton shower and also an extended comparison of our parton shower algorithm with jet and event shape data from the LEP experiments, from the Tevatron Run II measurements as well as first results from the 2010/11 LHC run. Without performing an overly sophisticated tuning of the shower, we reproduced the gross features of a big number of jet and event shape variables at lepton and hadron colliders and found in all cases good agreement.

This thesis is intended to serve as a proof of concept that an analytic parton shower for the initial state is viable to describe QCD in a realistic collider environment. Future lines of developments will contain a more extensive tuning and validation of the shower as well as the matching and merging prescription. We will also be investigating a possible exchange of the evolution variable for the transverse momentum, $p_T$, which would guarantee angular ordering and color coherence right from the beginning and might simplify or even improve on the parton shower description given in our algorithm. A development of an interleaved multiple interaction algorithm connected with a properly color-connected analytic initial-state parton shower together with its implementation is in preparation and will be part of a future publication. Moreover an implementation of a CKKW-type matching, giving the opportunity to choose between two implemented matching schemes, is planned for the near future.

The implementations of the analytic parton shower and the MLM matching will become publicly available with the upcoming release 2.1 of WHIZARD, scheduled to be published in June 2012.

# Acknowledgments

Many thanks go to my supervisor Jürgen Reuter for the opportunity to do my Ph.D. thesis. Further thanks go to Bernd Kniehl for acting as second examiner of this thesis and Jochen Bartels and Georg Steinbrck for taking part in the examination committee.

In addition, I would like to thank my office colleagues (in more or less chronological order) Felix Braam, Anke Butenuth, Kathrin Mallot, Daniel Wiesler, Enrico Rinaldi, Erik Gerwick, Michael Grefe, Jasper Hasenkamp, Kai Schmitz and Václav Tlapák for an enjoyable atmosphere in the office.

I would also like to thank Simon Plätzer and Daniel Wiesler for helpful discussions about parton showers and matching. Special thanks go to Felix Braam, Marco Tonini and Hans-Werner Boschmann for valuable comments on drafts of this thesis. Finally, many thanks go to the particle physics theory groups in Freiburg, Edinburgh and Hamburg for an enjoyable time during the years of preparing this thesis.

# A Observables

## A.1 Event Shape Variables

This section contains descriptions of the event shapes used inside this thesis. All event shapes shown are only applicable for lepton colliders, generalizations to hadronic collisions are also available, for example *beam-thrust* [95], but not used within this thesis.

### A.1.1 Thrust-like Variables

In the following, summations are always over all final state partons. The maximizations are always over all vectors $\mathbf{n}$ with length 1.

- Thrust $T$:
$$T = \max_{\mathbf{n}} \frac{\sum_i |\mathbf{p}_i \cdot \mathbf{n}|}{\sum_i |\mathbf{p}_i|} \tag{A.1}$$
Thrust lies in the range $[0.5 : 1]$, where $T = 1$ corresponds to a pencil-like event topology and $T = 0.5$ corresponds to a perfectly spherical event topology. The axis $\mathbf{n}_T$ that maximizes (A.1) is the axis of most activity.

- Thrust major $T_{maj}$:
$$T_{maj} = \max_{\mathbf{n},\mathbf{n}\cdot\mathbf{n}_T=0} \frac{\sum_i |\mathbf{p}_i \cdot \mathbf{n}|}{\sum_i |\mathbf{p}_i|}, \tag{A.2}$$
where $\mathbf{n}_T$ is the vector that maximizes the term for the thrust $T$. The definition of thrust major is equivalent to the definition of thrust, only the allowed region for the test-vector $\mathbf{n}$ is constrained to lie in a plane perpendicular to the thrust axis. Therefore thrust major is smaller than thrust. Thrust major therefore measures the pencil-likeness and sphericality in the plane perpendicular to $\mathbf{n}_T$. Like thrust, thrust major is confined to the range $[0 : 1]$, due to the constraint that thrust major has to be smaller than thrust, the range is reduced further. Thrust major lies in the range $[0 : 2/\pi]$[32].

- Thrust minor $T_{min}$:
$$T_{min} = \left.\frac{\sum_i |\mathbf{p}_i \cdot \mathbf{n}|}{\sum_i |\mathbf{p}_i|}\right|_{\mathbf{n}\cdot\mathbf{n}_T=\mathbf{n}\cdot\mathbf{n}_{T_{maj}}=0}, \tag{A.3}$$

[32] See section B.1

where $\mathbf{n}_T$ and $\mathbf{n}_{T_{maj}}$ are the vectors that maximize the terms for the thrust $T$ and the Thrust major $T_{maj}$, respectively. Thrust minor is a measure for the activity along the axis perpendicular to the axes of most activity, $\mathbf{n}_T$ and $\mathbf{n}_{T_{maj}}$. Its range is confined to $[0 : T_{maj}]$.

- Oblateness $O$:

$$O = T_{maj} - T_{min} \tag{A.4}$$

Oblateness is a measure for flatness. An event that has most of its activity in a plane will have the axes $\mathbf{n}_T$ and $\mathbf{n}_{T_{maj}}$ inside this plane, not leaving much activity perpendicular to this plane. Therefore $T_{min}$ will be small and the oblateness will be close to the thrust major. On the other hand, for an event that is rotationally symmetric around its axis of highest activity, $T_{maj}$ and $T_{min}$ will be of comparable size, leading to an oblateness near zero. These are the two extreme cases for oblateness, its range is $[0 : T_{maj}]$.

- Hemisphere broadenings $B$:

$$B_{\pm} = \frac{\sum\limits_{\pm\mathbf{p}_i \cdot \mathbf{n}_{Thrust} > 0} |\mathbf{p}_i \times \mathbf{n}_{Thrust}|}{2 \sum\limits_i |\mathbf{p}_i|}$$

$$B_{max} = \max(B_+, B_-) \qquad B_{min} = \min(B_+, B_-)$$
$$B_{sum} = B_+ + B_- \qquad B_{diff} = |B_+ - B_-|$$

The hemisphere broadenings measure the split-up of the activity inside the two hemispheres, given by the plane perpendicular to the thrust axis. The hemispheres are defined using the thrust axis, because not much activity is along this plane. For example an event with a pencil-like configuration in one hemisphere, but a more evenly distributed configuration in the other one, will give a rather high value for $B_{max}$ due to the hemisphere with the more evenly distribution, a low value for $B_{min}$ due to the pencil-like configuration in the other hemisphere and a high value for $B_{diff}$ due to the difference between the hemispheres.

The following table gives the aforementioned variables for some main event topologies:

| type of event | $T$ | $T_{maj}$ | $T_{min}$ | $O$ | $B_{max}$ | $B_{min}$ | $B_{sum}$ | $B_{diff}$ |
|---|---|---|---|---|---|---|---|---|
| pencil-like (2-jet) | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| symmetric 3-jet | 2/3 | $1/\sqrt{3}$ | 0 | $1/\sqrt{3}$ | $\sqrt{3}/6$ | 0 | $\sqrt{3}/6$ | $\sqrt{3}/6$ |
| spherical | 1/2 | 1/2 | 1/2 | 0 | $\pi/8$ | $\pi/8$ | $\pi/4$ | 0 |

For a nice introduction to thrust-like variables see [96], for more information about hemisphere broadenings, see [97]. The event shape variables are also abundantly used in the analyses presented by the DELPHI collaboration [86, 87] used for comparison in section 5.

### A.1.2 Energy-energy Correlation

The definition for the EEC [98] is given by

$$\begin{aligned} EEC(\cos\chi) &= \frac{1}{2N}\frac{1}{\Delta\cos\chi}\sum_{events}\sum_{i,j}(1-\delta_{ij})\frac{E_iE_j}{E_{vis}^2} \\ &\quad\times\Theta\left(\Delta\cos\chi-|\cos\chi-\cos\chi_{ij}|\right), \end{aligned} \tag{A.5}$$

where $\cos\chi$ and $\Delta\cos\chi$ are the lower edge and width of a bin, respectively, and $\Theta$ is the step function. The energy-energy correlation describes the distribution of angles $\cos\chi_{ij}$ scaled by their energies.

## A.2 Jet Algorithms

Jet algorithms are tools to organize the plethora of particles produced in a collision. This is done by grouping "similar" particles into one pseudo-particle called *jet*. The criteria can be the closeness in the geometry of the detector, leading to *cone-jet algorithms*, where all particles within a cone of "radius" $R$ are assumed to be one jet. The measure $R$ is given by $R=\sqrt{(\Delta\eta)^2+(\Delta\phi)^2}$ with the pseudo-rapidity $\eta$ and the azimuthal angle $\phi$. For a further discussion of the problems arising from this approach see e.g. SISCone [99]. A different approach is to sequentially combine two particles into a new pseudo-particle. The two particles are chosen by selecting the pair of partons for which a distance-measure $y_{ij}$ is the smallest. A common choice for this distance-measure is

$$y_{ij}=2\frac{\min\left(E_i,E_j\right)^2}{s}\left(1-\cos\theta_{ij}\right) \tag{A.6}$$

or, because $s$ is a constant, the division by $s$ can be omitted, resulting in the dimensionful version

$$y_{ij}=2\min\left(E_i,E_j\right)^2\left(1-\cos\theta_{ij}\right) \tag{A.7}$$

for $e^+e^-$-collisions and

$$y_{ij} = \left(\Delta R_{ij}\right)^2\min\left(p_{\perp i}^2,p_{\perp j}^2\right), \tag{A.8}$$

$$y_i = p_{\perp i}^2 \tag{A.9}$$

for hadronic collisions, yielding also the possibility to cluster partons to the beam-axis. These prescriptions for the distance-measures compose the so-called $k_T$-algorithm. The algorithm can be varied by replacing the 2 in the exponent in equation (A.6). Other values that have been studied are 0 and $-2$, changing to 0 leads to the *Cambridge-Aachen* algorithm [100, 101], while changing to $-2$ produces the *anti-$k_T$*-algorithm [102]. The anti-$k_T$-algorithm is the most used version at the ATLAS and CMS experiments.

# B Useful Formula

## B.1 Maximal Value for $T_{maj}$

The maximal value for Thrust major is obtained for a perfectly planar distribution. Let this distribution be

$$\mathbf{p}(\phi) = \frac{1}{2\pi}\left(\cos\phi, \sin\phi, 0\right) \tag{B.1}$$

with $\phi \in [0 : 2\pi)$. Then $\mathbf{n}_T$ and $\mathbf{n}_{T_{maj}}$ will lie in the $x$-$y$-plane, let them be $\mathbf{n}_T = (1, 0, 0)$ and $\mathbf{n}_{T_{maj}} = (0, 1, 0)$.

$$\begin{aligned}
T_{maj}^{max} &= \max_{\mathbf{n},\mathbf{n}\cdot\mathbf{n}_T=0} \frac{\sum_i |\mathbf{p}_i \cdot \mathbf{n}|}{\sum_i |\mathbf{p}_i|} && \text{(B.2)}\\
&= \max_{\mathbf{n},\mathbf{n}\cdot\mathbf{n}_T=0} \frac{\int_0^{2\pi} \mathrm{d}\phi |\mathbf{p}(\phi)\cdot\mathbf{n}|}{\int_0^{2\pi} \mathrm{d}\phi\ |\mathbf{p}(\phi)|} && \text{(B.3)}\\
&= \frac{\int_0^{2\pi} \mathrm{d}\phi\ \frac{1}{2\pi} |\sin\phi|}{\int_0^{2\pi} \mathrm{d}\phi\ \frac{1}{2\pi}} && \text{(B.4)}\\
&= \frac{2}{2\pi}\int_0^{\pi} \mathrm{d}\phi\ \sin\phi && \text{(B.5)}\\
&= \frac{2}{\pi} \approx 0,63 && \text{(B.6)}
\end{aligned}$$

Thus for an evenly distributed planar distribution the values for thrust and thrust major are

$$T = T_{maj} = \frac{2}{\pi}. \tag{B.7}$$

## B.2 Bounds on $z$

### B.2.1 Minimum Transverse momentum

Let $|p_{CM}|$" be the magnitude of the daughters' three-momenta in the rest frame of the mother and $\theta$ be the angle between the direction of the mother's momentum in the lab-frame and the first daughter's momentum in the rest-frame of the mother, just like in the formulation of the analytic FSR. Then, the transverse momentum with respect to the mother's flight direction is given by

$$p_T = |p_{CM}|\sqrt{1-\cos^2\theta}, \tag{B.8}$$

which can be transformed, assuming massless daughters and using equations (6), (7) and (10) from [47],

$$= \sqrt{\frac{t_0}{4}}\sqrt{1-\frac{4(z-\frac{1}{2})^2}{\beta^2}} \tag{B.9}$$

$$= \sqrt{\frac{t_0}{4}-\frac{t_0}{\beta^2}\left(z-\frac{1}{2}\right)^2} \tag{B.10}$$

$$p_T^2 = \frac{t_0}{4}-\frac{t_0}{\beta^2}\left(z-\frac{1}{2}\right)^2 \tag{B.11}$$

The cut $p_T^2 \geq \frac{t_{cut}}{4}$ thus transforms to

$$\left|z-\frac{1}{2}\right| \leq \frac{\beta}{2}\sqrt{1-\frac{t_0}{t_{cut}}}. \tag{B.12}$$

### B.2.2 Color coherence

The opening angle $\cos\vartheta$ between the two (massless) partons produced in a branching of a parton at virtuality $t$ is given by

$$\cos\vartheta = \frac{1}{2|\mathbf{p}_1||\mathbf{p}_1|}\left(p_1^2+p_2^2+2E_1E_2-t\right). \tag{B.13}$$

Using $p_1^2 = p_2^2 = 0$, $E_1 = zE, E_2 = (1-z)E$ and $|\mathbf{p}_i| = E_i$, this gives

$$\cos\vartheta = 1-\frac{t}{2z(1-z)E^2} \tag{B.14}$$

$$= 1-\frac{t}{2\left(-|z-\frac{1}{2}|^2+\frac{1}{4}\right)E^2} \tag{B.15}$$

Now the constraint on the opening angle leads to

$$\cos\vartheta_{cut} \leq \cos\vartheta \tag{B.16}$$

$$\cos\vartheta_{cut} \leq 1 - \frac{t}{2\left(-|z-\frac{1}{2}|^2+\frac{1}{4}\right)E^2} \tag{B.17}$$

$$\frac{t}{2E^2}\frac{1}{1-\cos\vartheta_{cut}} \leq -\left|z-\frac{1}{2}\right|^2 + \frac{1}{4} \tag{B.18}$$

$$\left|z-\frac{1}{2}\right|^2 \leq \frac{1}{4} - \frac{t}{2E^2}\frac{1}{1-\cos\vartheta_{cut}} \tag{B.19}$$

Using $\frac{1}{1-\cos\vartheta_{cut}} = \frac{1}{2}\left[\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}}+1\right]$ leads to

$$\left|z-\frac{1}{2}\right|^2 \leq \frac{1}{4} - \frac{t}{4E^2}\left[\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}}+1\right] \tag{B.20}$$

$$\left|z-\frac{1}{2}\right|^2 \leq \frac{1}{4}\left[1-\frac{t}{E^2}\right] - \frac{t}{4E^2}\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}} \tag{B.21}$$

$$\left|z-\frac{1}{2}\right|^2 \leq \frac{\beta^2}{4} - \frac{t}{4E^2}\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}} \tag{B.22}$$

$$\left|z-\frac{1}{2}\right|^2 \leq \frac{\beta^2}{4}\left[1-\frac{t}{\beta^2E^2}\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}}\right] \tag{B.23}$$

Taking the square root of both sides gives equation (3.13):

$$\left|z-\frac{1}{2}\right| \leq \frac{\beta}{2}\sqrt{1-\frac{t}{\beta^2E^2}\frac{1+\cos\vartheta_{cut}}{1-\cos\vartheta_{cut}}}. \tag{B.24}$$

This cut was first introduced in equation (26) in [47].

# C Sample Source Code

## C.1 Sample Standalone Main Program

The following program simulates the parton shower for one event for the process $e^+e^- \to d\bar{d}$ at a center-of-mass energy of $\sqrt{\hat{s}} = 91\,\text{GeV}$. For simplicity the unshowered quarks are assumed to travel along the $z$-axis. The generated event is printed to the screen.

```
program main

  use shower_module
  use kinds, only: double

  implicit none

  type(shower_t) :: shower
  type(parton_t), pointer :: prt1, prt2, prt3, prt4
  type(parton_pointer_t), dimension(:),
                          allocatable :: particles
  real(kind=double) :: shat = 91.0**2

  call shower_create(shower)

  ! allocate 4 partons: 2 incoming, 2 outgoing
  allocate(prt1)
  allocate(prt2)
  allocate(prt3)
  allocate(prt4)

  ! assign e+ e- => d dbar
  prt1%nr=shower_get_next_free_nr(shower)
  prt1%typ=11
  call parton_set_momentum(prt1, 0.5_double*sqrt(shat),
           0._double, 0._double, 0.5_double*sqrt(shat))
  prt2%nr=shower_get_next_free_nr(shower)
  prt2%typ=-11
  call parton_set_momentum(prt2, 0.5_double*sqrt(shat),
           0._double, 0._double,-0.5_double*sqrt(shat))
  prt3%nr=shower_get_next_free_nr(shower)
  prt3%typ=1
  call parton_set_momentum(prt3, 0.5_double*sqrt(shat),
           0._double, 0._double, 0.5_double*sqrt(shat))
  prt4%nr=shower_get_next_free_nr(shower)
  prt4%typ=-1
  call parton_set_momentum(prt4, 0.5_double*sqrt(shat),
           0._double, 0._double,-0.5_double*sqrt(shat))
  ! assign color connection
  prt3%c1=shower_get_next_color_nr(shower)
  prt4%c2=prt3%c1
```

```
  ! add this new "interaction" to the shower
  allocate(particles(1:4))
  particles(1)%p=>prt1
  particles(2)%p=>prt2
  particles(3)%p=>prt3
  particles(4)%p=>prt4
  call shower_add_interaction2ton(shower, particles)

  ! perform the fsr and print the event
   call shower_interaction_generate_fsr2ton(shower,
                              shower%interactions(1)%i)
   call shower_print(shower)

   ! clean up
   call shower_final(shower)
   deallocate(particles)
   deallocate(prt1)
   deallocate(prt2)
   deallocate(prt3)
   deallocate(prt4)

end program main
```

## C.2 Sample SINDARIN Files

SINDARIN is the scripting language designed for WHIZARD's input files. For more information about SINDARIN, see the WHIZARD manual or the recent publication about WHIZARD [29]. The following SINDARIN examples can also be found among the examples provided in trunk/share/examples in WHIZARD's source code repository.

### C.2.1 Final-State Radiation

The following example simulates the process $e^+e^- \to u\bar{u}$ at a center-of-mass energy of $\sqrt{s} = 91\,\text{GeV}$. Both, FSR and ISR, are enabled, although there is no initial-state radiation to be simulated. The cross section for the process is calculated. Then, 100,000 events are simulated and written to a Les Houches Event file.

```
model = SM
```

```
process eeuu = e1, E1 => u, U

compile

sqrts = 91 GeV
beams = e1, E1

?ps_fsr_active=true
?ps_isr_active=true
?hadronization_active = true
?ps_use_PYTHIA_shower = false

ps_max_n_flavors = 5
ps_mass_cutoff = 1
ps_fsr_lambda = 0.29

?mlm_matching = false

integrate(eeuu) { iterations = 2:2000 }

n_events = 100000

?rebuild_events=true
$sample = "EENoMatchingW"
sample_format = lhef

simulate(eeuu)

show(results)
```

### C.2.2 Final-State Radiation with Matching

This example is similar to the previous one. The process $e^+e^- \to u\bar{u}$ is supplemented with the two additional higher contributions, $e^+e^- \to u\bar{u}j$ and $e^+e^- \to u\bar{u}jj$, where $j$ is any of the $u, d, s, c$-(anti-)quarks or a gluon. This is the configuration for the MLM matching with two additional jets. Thus the additional cuts on the matrix elements are necessary.

```
model = SM
ms = 0
mc = 0
```

```
alias j = u:d:s:c:U:D:S:C:g
process eeuu = e1, E1 => u, U
process eeuuj = e1, E1 => u, U, j
process eeuujj = e1, E1 => u, U, j, j

compile

sqrts = 91 GeV
beams = e1, E1

?ps_fsr_active=true
?ps_isr_active=true
?hadronization_active = true
?ps_use_PYTHIA_shower = false

ps_max_n_flavors = 5
ps_mass_cutoff = 1
ps_fsr_lambda = 0.29

?mlm_matching = true
mlm_ptmin = 10 GeV
mlm_nmaxMEjets = 4
mlm_Rmin = 1
mlm_Emin = 10 GeV

cuts = all E > mlm_Emin [j]
       and all Dist > mlm_Rmin [j, j]

integrate(eeuu, eeuuj, eeuujj) { iterations = 2:2000 }

n_events = 100000

?rebuild_events=true
$sample = "EEMatching4W"
sample_format = lhef

simulate(eeuu, eeuuj, eeuujj)

show(results)
```

### C.2.3 Initial-State Radiation

The following SINDARIN file simulates Drell-Yan production at the Tevatron, $p\bar{p} \to Z + X$, via the process $q\bar{q} \to e^+e^-$. The LHAPDF library is used for the parton distribution functions, the cuts are used to constrain the matrix element to a region around the $Z$-pole. The $p_T$-distribution of the $Z$, or the $e^+e^-$-pair, respectively, in the 50,000 simulated events is recorded in a histogram.

```
model = SM
alias quark = u:d:s:c
alias antiq = U:D:S:C
process qqee = quark:antiq, quark:antiq => e1, E1

ms = 0
mc = 0

compile

sqrts = 1960 GeV
beams = p, pbar => lhapdf

$title = "Pt of Z"
$x_label = "Pt / GeV"
$y_label = "N"
histogram Pt_distribution (0, 250, 1)

ps_isr_primordial_kt_width = 0
ps_isr_tscalefactor = 1

?ps_fsr_active=true
?ps_isr_active=true
?hadronization_active = true
?ps_use_PYTHIA_shower = false
ps_max_n_flavors = 5
ps_mass_cutoff = 0.5

!! Rick Field's CDF Tune D6 using CTEQ6L1
$ps_PYTHIA_PYGIVE = "MSTP(5)=108"

?mlm_matching = false
```

```
cuts = all M > 80 GeV [combine[e1,E1]]
       and all M < 100 GeV [combine[e1,E1]]

integrate(qqee) { iterations = 2:20000 }

n_events = 50000

?rebuild_events=true
$sample = "DrellYanNoMatchingW"
sample_format = lhef

analysis = record Pt_distribution (eval Pt [combine[e1,E1]])

simulate(qqee)

show(results)
```

### C.2.4 Initial-State Radiation with Matching

This is the version of the previous SINDARIN file with MLM matching. The 0-jet process $q\bar{q} \to e^+e^-$ is supplemented by the gluon emission process $q\bar{q} \to e^+e^-g$ and the two processes evolving from the equivalent initial states, $qg \to e^+e^-q$ and $\bar{q}g \to e^+e^-\bar{q}$.

```
model = SM
alias quark = u:d:s:c
alias antiq = U:D:S:C
process qqee = quark:antiq, quark:antiq => e1, E1
process qqeeg = quark:antiq, quark:antiq => e1, E1, g
process qgeeq = quark:antiq, g => e1, E1, quark:antiq
process gqeeq = g, quark:antiq => e1, E1, quark:antiq

ms = 0
mc = 0

compile

sqrts = 1960 GeV
beams = p, pbar => lhapdf

$title = "Pt of Z"
```

```
$x_label = "Pt / GeV"
$y_label = "N"
histogram Pt_distribution (0, 250, 1)

ps_isr_primordial_kt_width = 0
ps_isr_tscalefactor = 1

?ps_fsr_active=true
?ps_isr_active=true
?hadronization_active = true
?ps_use_PYTHIA_shower = false
ps_max_n_flavors = 5
ps_mass_cutoff = 0.5

!! MSTP(5)=108 : Rick Field's CDF Tune D6 using CTEQ6L1
!! MSTP(81)=0  : no multiple interactions
!! MSTJ(41)=1  : only QCD branchings
!! MSTJ(21)=0  : no decays
!! MSTP(68)=0  : turn off PYTHIA's matching
$ps_PYTHIA_PYGIVE = "MSTP(5)=108;MSTP(81)=0;MSTJ(41)=1;
                     MSTJ(21)=0;MSTP(68)=0"

?mlm_matching = true
mlm_ptmin = 5 GeV
mlm_etamax = 2.5
mlm_Rmin = 1
mlm_nmaxMEjets = 1

cuts = all M > 80 GeV [combine[e1,E1]]
       and all M < 100 GeV [combine[e1,E1]]
       and all Pt > mlm_ptmin [g:quark:antiq]
       and all abs(Eta) < mlm_etamax [g:quark:antiq]

integrate(qqee, qqeeg, qgeeq, gqeeq) { iterations = 2:20000 }

n_events = 50000

?rebuild_events=true
$sample = "DrellYanMatchingW"
sample_format = lhef
```

```
analysis = record Pt_distribution (eval Pt [combine[e1,E1]])

simulate(qqee, qqeeg, qgeeq, gqeeq)

show(results)
```

# D ATLAS Pile-up Measurements

The figures D.1 and D.2 show the measurements of pile-up events performed at the ATLAS detector. The images were taken from https://twiki.cern.ch/twiki/bin/view/AtlasPublic/LuminosityPublicResults.

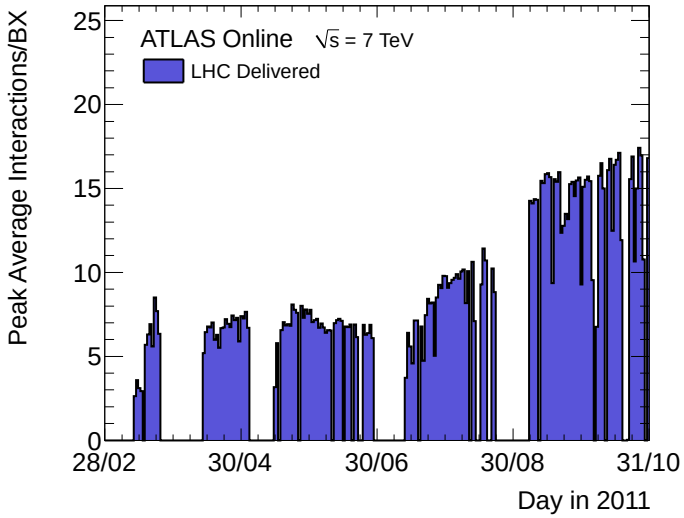


Figure D.1: Average number of proton-proton interactions per bunch crossing as measured in the ATLAS detector.

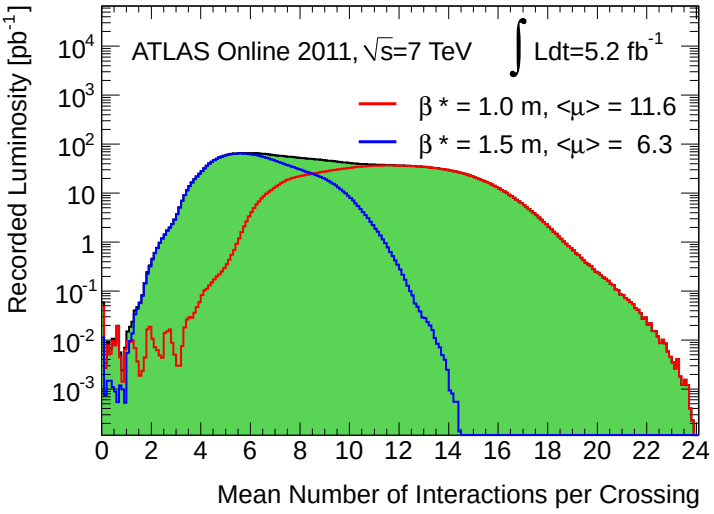


Figure D.2: Distribution of the number of pile-up events as measured in the ATLAS detector. The blue and red curves correspond to data taken before and after a technical stop in September.

# Bibliography

[1] Sebastian Schmidt. Ein analytischer Parton Shower für Initial State Radiation. Master's thesis, Universität Siegen, 2008.

[2] F.J. Hasert et al. Observation of Neutrino Like Interactions Without Muon Or Electron in the Gargamelle Neutrino Experiment. *Phys.Lett.*, B46:138–140, 1973.

[3] J.J. Aubert et al. Experimental Observation of a Heavy Particle J. *Phys.Rev.Lett.*, 33:1404–1406, 1974. Technical report 96.

[4] J.E. Augustin et al. Discovery of a Narrow Resonance in e+ e- Annihilation. *Phys.Rev.Lett.*, 33:1406–1408, 1974.

[5] G. Arnison et al. Experimental Observation of Isolated Large Transverse Energy Electrons with Associated Missing Energy at s**(1/2) = 540-GeV. *Phys.Lett.*, B122:103–116, 1983.

[6] G. Arnison et al. Experimental Observation of Lepton Pairs of Invariant Mass Around 95-GeV/c**2 at the CERN SPS Collider. *Phys.Lett.*, B126:398–410, 1983.

[7] S. Abachi et al. Search for high mass top quark production in $p\bar{p}$ collisions at $\sqrt{s} = 1.8$ TeV. *Phys.Rev.Lett.*, 74:2422–2426, 1995, hep-ex/9411001.

[8] F. Abe et al. Observation of top quark production in $\bar{p}p$ collisions. *Phys.Rev.Lett.*, 74:2626–2631, 1995, hep-ex/9503002.

[9] K. Kodama et al. Observation of tau neutrino interactions. *Phys.Lett.*, B504:218–224, 2001, hep-ex/0012035.

[10] K. Nakamura et al. Review of particle physics. *J. Phys.*, G37:075021, 2010.

[11] Eric Laenen. Resummation for observables at TeV colliders. *Pramana*, 63:1225–1249, 2004. 10.1007/BF02704892.

[12] S. Durr, Z. Fodor, J. Frison, C. Hoelbling, R. Hoffmann, et al. Ab-Initio Determination of Light Hadron Masses. *Science*, 322:1224–1227, 2008, 0906.3599.

[13] Rick Field. Min-Bias and the Underlying Event at the LHC. *Acta Phys.Polon.*, B42:2631–2656, 2011, 1110.5530.

[14] J.R. Andersen et al. The SM and NLO Multileg Working Group: Summary report. pages 21–189, 2010, 1003.1241.

[15] Paolo Nason and Bryan Webber. Next-to-Leading-Order Event Generators. 2012, 1202.1251.

[16] J. Alcaraz Maestre et al. The SM and NLO Multileg and SM MC Working Groups: Summary Report. 2012, 1203.6803.

[17] F. Bloch and A. Nordsieck. Note on the Radiation Field of the electron. *Phys.Rev.*, 52:54–59, 1937.

[18] T. Kinoshita. Mass singularities of Feynman amplitudes. *J.Math.Phys.*, 3:650–677, 1962.

[19] T.D. Lee and M. Nauenberg. Degenerate Systems and Mass Singularities. *Phys.Rev.*, 133:B1549–B1562, 1964.

[20] Martin Lavelle and David McMullan. Collinearity, convergence and cancelling infrared divergences. *JHEP*, 0603:026, 2006, hep-ph/0511314.

[21] Carl D. Anderson. The positive electron. *Phys. Rev.*, 43(6):491–494, Mar 1933.

[22] ATLAS detector and physics performance. Technical design report. Vol. 2. CERN-LHCC-99-15.

[23] Torbjörn Sjöstrand, Stephen Mrenna, and Peter Z. Skands. PYTHIA 6.4 Physics and Manual. *JHEP*, 05:026, 2006, hep-ph/0603175.

[24] Torbjörn Sjöstrand, Stephen Mrenna, and Peter Z. Skands. A Brief Introduction to PYTHIA 8.1. *Comput.Phys.Commun.*, 178:852–867, 2008, 0710.3820.

[25] Michelangelo L. Mangano, Mauro Moretti, Fulvio Piccinini, Roberto Pittau, and Antonio D. Polosa. ALPGEN, a generator for hard multiparton processes in hadronic collisions. *JHEP*, 0307:001, 2003, hep-ph/0206293.

[26] M. Bahr et al. Herwig++ Physics and Manual. *Eur. Phys. J.*, C58:639–707, 2008, 0803.0883.

[27] Johan Alwall, Michel Herquet, Fabio Maltoni, Olivier Mattelaer, and Tim Stelzer. MadGraph 5 : Going Beyond. *JHEP*, 1106:128, 2011, 1106.0522.

[28] T. Gleisberg et al. Event generation with SHERPA 1.1. *JHEP*, 02:007, 2009, 0811.4622.

[29] Wolfgang Kilian, Thorsten Ohl, and Jürgen Reuter. WHIZARD: Simulating Multi-Particle Processes at LHC and ILC. *Eur.Phys.J.*, C71:1742, 2011, 0708.4233.

[30] M. R. Whalley, D. Bourilkov, and R. C. Group. The Les Houches Accord PDFs (LHAPDF) and Lhaglue. 2005, hep-ph/0508110.

[31] Neil D. Christensen and Claude Duhr. FeynRules - Feynman rules made easy. *Comput.Phys.Commun.*, 180:1614–1641, 2009, 0806.4194.

[32] Hung-Liang Lai, Marco Guzzi, Joey Huston, Zhao Li, Pavel M. Nadolsky, et al. New parton distributions for collider physics. *Phys.Rev.*, D82:074024, 2010, 1007.2241.

[33] A.D. Martin, W.J. Stirling, R.S. Thorne, and G. Watt. Parton distributions for the LHC. *Eur.Phys.J.*, C63:189–285, 2009, 0901.0002.

[34] Richard D. Ball, Luigi Del Debbio, Stefano Forte, Alberto Guffanti, Jose I. Latorre, et al. A first unbiased global NLO determination of parton distributions and their uncertainties. *Nucl.Phys.*, B838:136–206, 2010, 1002.4407.

[35] Richard Corke and Torbjörn Sjöstrand. Multiparton Interactions and Rescattering. *JHEP*, 1001:035, 2010, 0911.1909.

[36] Torbjörn Sjöstrand and Peter Z. Skands. Transverse-momentum-ordered showers and interleaved multiple interactions. *Eur.Phys.J.*, C39:129–154, 2005, hep-ph/0408302.

[37] Hans-Werner Boschmann. *Mehrfache Partonische Wechselwirkungen bei Hadronischen Streuungen*. PhD thesis, Universität Siegen, 2011.

[38] Jonathan R. Gaunt and W.James Stirling. Double Parton Distributions Incorporating Perturbative QCD Evolution and Momentum and Quark Number Sum Rules. *JHEP*, 1003:005, 2010, 0910.4347.

[39] Bo Andersson, G. Gustafson, G. Ingelman, and T. Sjostrand. Parton Fragmentation and String Dynamics. *Phys. Rept.*, 97:31–145, 1983.

[40] G. Marchesini and B.R. Webber. Monte Carlo Simulation of General Hard Processes with Coherent QCD Radiation. *Nucl.Phys.*, B310:461, 1988.

[41] S. Albino, F. Anulli, F. Arleo, Dave Z. Besson, William K. Brooks, et al. Parton fragmentation in the vacuum and in the medium. 2008, 0804.2021.

[42] G. Aad et al. Expected Performance of the ATLAS Experiment - Detector, Trigger and Physics. 2009, 0901.0512.

[43] G. L. Bayatian et al. CMS physics: Technical design report. CERN-LHCC-2006-001.

[44] Leif Lonnblad. ARIADNE version 4: A Program for simulation of QCD cascades implementing the color dipole model. *Comput. Phys. Commun.*, 71:15–31, 1992.

[45] Walter T. Giele, David A. Kosower, and Peter Z. Skands. A Simple shower and matching algorithm. *Phys.Rev.*, D78:014026, 2008, 0707.3652.

[46] Christian W. Bauer and Matthew D. Schwartz. Event generation from effective field theory. *PHYS.REV.D*, 76:074004, 2007.

[47] Christian W. Bauer and Frank J. Tackmann. Gaining analytic control of parton showers. *Phys. Rev.*, D76:114017, 2007, 0705.1719.

[48] Christian W. Bauer, Frank J. Tackmann, and Jesse Thaler. GenEvA (II): A phase space generator from a reweighted parton shower. *JHEP*, 12:011, 2008, 0801.4028.

[49] Wolfgang Kilian, Jürgen Reuter, Sebastian Schmidt, and Daniel Wiesler. An Analytic Initial-State Parton Shower. *JHEP*, 1204:013, 2012, 1112.1039.

[50] Philip Stephens and Andre van Hameren. Propagation of uncertainty in a parton shower. 2007, hep-ph/0703240.

[51] W. T. Giele, D. A. Kosower, and P. Z. Skands. Higher-Order Corrections to Timelike Jets. *Phys. Rev.*, D84:054003, 2011, 1102.2126.

[52] Manfred Böhm, Ansgar Denner, and Hans Joos. *Gauge Theories of the Strong and Electroweak Interaction.* Teubner Verlag, 4 2001.

[53] R. K. Ellis, W. J. Stirling, and B. R. Webber. *QCD and Collider Physics.* Cambridge University Press, 1986.

[54] Francesco Caravaglios and Mauro Moretti. An algorithm to compute Born scattering amplitudes without Feynman graphs. *Phys.Lett.*, B358:332–338, 1995, hep-ph/9507237.

[55] Mauro Moretti, Thorsten Ohl, and Jurgen Reuter. O'Mega: An Optimizing matrix element generator. 2001, hep-ph/0102195.

[56] Aggeliki Kanaki and Costas G. Papadopoulos. HELAC: A Package to compute electroweak helicity amplitudes. *Comput.Phys.Commun.*, 132:306–315, 2000, hep-ph/0002082. 14 pages Report-no: DEMO-HEP-2000/01.

[57] Alessandro Cafarella, Costas G. Papadopoulos, and Malgorzata Worek. Helac-Phegas: A Generator for all parton level processes. *Comput.Phys.Commun.*, 180:1941–1955, 2009, 0710.2427.

[58] V.V. Sudakov. Vertex parts at very high-energies in quantum electrodynamics. *Sov.Phys.JETP*, 3:65–71, 1956.

[59] D. Amati, A. Bassetto, M. Ciafaloni, G. Marchesini, and G. Veneziano. A Treatment of Hard Processes Sensitive to the Infrared Structure of QCD. *Nucl.Phys.*, B173:429, 1980.

[60] A. Bassetto, M. Ciafaloni, and G. Marchesini. Jet Structure and Infrared Sensitive Quantities in Perturbative QCD. *Phys.Rept.*, 100:201–272, 1983.

[61] F. Abe et al. Evidence for color coherence in $p\bar{p}$ collisions at $\sqrt{s} = 1.8$ TeV. *Phys.Rev.*, D50:5562–5579, 1994.

[62] A.E. Chudakov. On an ionization effect related to the observation of electronpositron pairs at very high energies. *Bulletin of the Academy of Sciences of the USSR. Physical Series*, 19(6):589–595, 1955.

[63] Torbjorn Sjostrand. A Model for Initial State Parton Showers. *Phys.Lett.*, B157:321, 1985.

[64] V. N. Gribov and L. N. Lipatov. Deep inelastic e p scattering in perturbation theory. *Sov. J. Nucl. Phys.*, 15:438–450, 1972.

[65] Guido Altarelli and G. Parisi. Asymptotic Freedom in Parton Language. *Nucl. Phys.*, B126:298, 1977.

[66] Yuri L. Dokshitzer. Calculation of the Structure Functions for Deep Inelastic Scattering and e+ e- Annihilation by Perturbation Theory in Quantum Chromodynamics. *Sov. Phys. JETP*, 46:641–653, 1977.

[67] Nils Lavesson and Leif Lönnblad. Merging parton showers and matrix elements – back to basics. *JHEP*, 04:085, 2008, 0712.2966.

[68] S. Catani, Frank Krauss, R. Kuhn, and B. R. Webber. QCD Matrix Elements + Parton Showers. *JHEP*, 11:063, 2001, hep-ph/0109231.

[69] F. Krauss. Matrix elements and parton showers in hadronic interactions. *JHEP*, 08:015, 2002, hep-ph/0205283.

[70] Leif Lönnblad. Correcting the colour-dipole cascade model with fixed order matrix elements. *JHEP*, 05:046, 2002, hep-ph/0112284.

[71] Michelangelo Mangano. Merging multijet ME's with shower MC's: some studies of systematics, 2002. Talk given at ME/MC Tuning WG Meeting, Fermilab, Batavia, U.S.A., 15. November 2002.

[72] Michelangelo Mangano, Mauro Moretti, and Roberto Pittau. Multijet matrix elements and shower evolution in hadronic collisions: W b anti-b + n jets as a case study. *Nucl. Phys.*, B632:343–362, 2002, hep-ph/0108069.

[73] Stefan Hoeche, Frank Krauss, Nils Lavesson, Leif Lönnblad, Michelangelo Mangano, et al. Matching parton showers and matrix elements. 2006, hep-ph/0602031.

[74] Johan Alwall et al. Comparative study of various algorithms for the merging of parton showers and matrix elements in hadronic collisions. *Eur. Phys. J.*, C53:473–500, 2008, 0706.2569.

[75] Stefano Frixione and Bryan R. Webber. Matching NLO QCD computations and parton shower simulations. *JHEP*, 0206:029, 2002, hep-ph/0204244.

[76] Paolo Nason. A New method for combining NLO QCD with shower Monte Carlo algorithms. *JHEP*, 0411:040, 2004, hep-ph/0409146.

[77] Stefano Frixione, Paolo Nason, and Carlo Oleari. Matching NLO QCD computations with Parton Shower simulations: the POWHEG method. *JHEP*, 0711:070, 2007, 0709.2092.

[78] S. Catani and M.H. Seymour. A General algorithm for calculating jet cross-sections in NLO QCD. *Nucl.Phys.*, B485:291–419, 1997, hep-ph/9605323.

[79] Stefano Catani, Stefan Dittmaier, Michael H. Seymour, and Zoltan Trocsanyi. The Dipole formalism for next-to-leading order QCD calculations with massive partons. *Nucl.Phys.*, B627:189–265, 2002, hep-ph/0201036.

[80] Gavin Cullen, Nicolas Greiner, Gudrun Heinrich, Gionata Luisoni, Pierpaolo Mastrolia, et al. Automated One-Loop Calculations with GoSam. 2011, 1111.2034.

[81] Christian W. Bauer, Frank J. Tackmann, and Jesse Thaler. GenEvA (I): A New framework for event generation. *JHEP*, 12:010, 2008, 0801.4026.

[82] L. Garren, C.-J. Lin, S. Navas, P. Richardson, T. Sjöstrand, and T. Trippe. 34. monte carlo particle numbering scheme, November 2009.

[83] Johan Alwall, A. Ballestrero, P. Bartalini, S. Belov, E. Boos, et al. A Standard format for Les Houches event files. *Comput.Phys.Commun.*, 176:300–304, 2007, hep-ph/0609017.

[84] Thorsten Ohl. Vegas revisited: Adaptive Monte Carlo integration beyond factorization. *Comput.Phys.Commun.*, 120:13–19, 1999, hep-ph/9806432.

[85] S. Catani, Yuri L. Dokshitzer, M.H. Seymour, and B.R. Webber. Longitudinally invariant $K_t$ clustering algorithms for hadron hadron collisions. *Nucl.Phys.*, B406:187–224, 1993.

[86] P. Abreu et al. Tuning and test of fragmentation models based on identified particles and precision event shape data. *Z.Phys.*, C73:11–60, 1996.

[87] P. Abreu et al. Measurement of event shape and inclusive distributions at S**(1/2) = 130-GeV and 136-GeV. *Z.Phys.*, C73:229–242, 1997.

[88] Andy Buckley, Jonathan Butterworth, Leif Lonnblad, Hendrik Hoeth, James Monk, et al. Rivet user manual. 2010, 1003.0694.

[89] P. Pfeifenschneider et al. QCD analyses and determinations of alpha(s) in e+ e- annihilation at energies between 35-GeV and 189-GeV. *Eur.Phys.J.*, C17:19–51, 2000, hep-ex/0001055.

[90] Simon Platzer and Stefan Gieseke. Dipole Showers and Automated NLO Matching in Herwig++. 2011, 1109.6256.

[91] Simon Plätzer. *Parton Showers and Radiative Corrections in QCD*. PhD thesis, Karlsruhe Institute of Technology, 2010.

[92] Gabriela Miu and Torbjorn Sjostrand. $W$ production in an improved parton shower approach. *Phys. Lett.*, B449:313–320, 1999, hep-ph/9812455.

[93] Victor Mukhamedovich Abazov et al. Measurement of the normalized $Z/\gamma^* \to \mu^+\mu^-$ transverse momentum distribution in $p\bar{p}$ collisions at $\sqrt{s} = 1.96$ TeV. *Phys.Lett.*, B693:522–530, 2010, 1006.0618.

[94] Serguei Chatrchyan et al. Measurement of the Rapidity and Transverse Momentum Distributions of Z Bosons in pp Collisions at sqrt(s)=7 TeV. *Phys. Rev. D 85,*, 032002, 2012, 1110.4973.

[95] Iain W. Stewart, Frank J. Tackmann, and Wouter J. Waalewijn. The Quark Beam Function at NNLL. *JHEP*, 1009:005, 2010, 1002.2213.

[96] S. Brandt and H. D. Dahmen. Axes and scalar measures of two-jet and three-jet events. *Zeitschrift für Physik C Particles and Fields*, 1:61–70, 1979. 10.1007/BF01450381.

[97] S. Catani, G. Turnock, and B.R. Webber. Jet broadening measures in $e^+e^-$ annihilation. *Phys.Lett.*, B295:269–276, 1992.

[98] C. Louis Basham, Lowell S. Brown, S.D. Ellis, and S.T. Love. Electron - Positron Annihilation Energy Pattern in Quantum Chromodynamics: Asymptotically Free Perturbation Theory. *Phys.Rev.*, D17:2298, 1978.

[99] Gavin P. Salam. A Practical Seedless Infrared Safe Cone Algorithm. 2007, 0705.2696.

[100] Yuri L. Dokshitzer, G.D. Leder, S. Moretti, and B.R. Webber. Better jet clustering algorithms. *JHEP*, 9708:001, 1997, hep-ph/9707323.

[101] M. Wobisch and T. Wengler. Hadronization corrections to jet cross-sections in deep inelastic scattering. 1998, hep-ph/9907280.

[102] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. The Anti-k(t) jet clustering algorithm. *JHEP*, 0804:063, 2008, 0802.1189.