CuTe 2.0.0
August 9, 2016

CuTe Manual

Thomas Becher[a], Thomas Lübbert[b], Matthias Neubert[c] and Daniel Wilhelm[c]

[a] *Institut für Theoretische Physik, Universität Bern*
Sidlerstrasse 5, CH–3012 Bern, Switzerland

[b] *II. Institut für Theoretische Physik, Universität Hamburg*
Luruper Chaussee 149, 22761 Hamburg, Germany

[c] *PRISMA Cluster of Excellence & Institut für Physik (THEP)*
Johannes Gutenberg-Universität, D–55099 Mainz, Germany

Abstract

CuTe is a C^{++} program and library combination to calculate the cross-sections for electro-weak gauge boson and Higgs boson production, differential in the boson's transverse momentum q_T and optionally also its pseudo-rapidity y. Using q_T resummation and a matching procedure to fixed order (FO), CuTe achieves at the same time next-to-next-to-next-to leading logarithmic (N^3LL) accuracy [1] in the resummation and next-to-next-to leading order (N^2LO) accuracy in the FO. Via the LHAPDF 6.x interface [2], it straightforwardly provides all supported parton distribution function (PDF) sets. Using the *Cuba library for multidimensional integration* (Cuba) [3] it is numerically highly accurate and efficient.

Contents

1 Introduction

CuTe is a program and library combination to compute the transverse momentum distribution of processes of the Drell-Yan (DY) type [**?**]. Its characteristic features are hadronic initial and color-neutral final states. It is fully implemented for photon, W- and Z-boson production via quark-antiquark annihilation and Higgs production via gluon-fusion.
The transverse-momentum distribution of DY-like processes is one of the most basic observables at hadron colliders. It is used e.g. to extract the W-boson mass and width and is of great phenomenological relevance for Higgs-production at the LHC. Especially the regime of small transverse-momentum $q_T^2 \ll q^2$ is important, because it gives the largest contribution to the total cross-section and is sensitive to *non-perturbative effects* (NPE). Here q_T denotes the transverse component of the boson momentum q and q^2 its invariant mass.
The hierarchy between the hard scale q^2 and the collinear scale q_T^2 leads to large logarithms $L_\lambda = \ln \frac{q_T^2}{q^2}$, which spoil the perturbativity of fixed-order (FO) calculations and need to be resummed to all orders. Our approach – based on *soft-collinear effective theory* (SCET) [4, 5] – was developed for the Drell-Yan process in [6], extended to higher orders and electroweak gauge-boson production in [7] and applied to Higgs-boson production in [8]. Using the N^{2}LO results [9, 10, 11] for the *transverse-distance dependant parton distribution functions* (TDPDF), we have further improved our results to ϵ^5+N^{3}LL+N^{2}LO precision [1].
Counting only powers of α_s beyond the Born level (W/Z $\sim \alpha_{em}$, DY $\sim \alpha_{em}^2$, Higgs $\sim \alpha_s^2$), this corresponds to the following precision: N^{3}LL denotes the resummation of all terms up to $\alpha_s^2(\alpha_s L_\lambda)^n$. This is the same as $\mathcal{O}\left(\alpha_s^2\right)$ accuracy in a scheme where one counts $L_\lambda \sim 1/\alpha_s$. N^{2}LO corresponds to the accuracy of the fixed-order expansion of our result to be $\mathcal{O}\left(\alpha_s^2\right)$. This accuracy is established by a matching procedure and leads to accurate predictions also at larger q_T values. In addition, we resum logarithms $L_\perp$ [1] – using an improved *power-counting* (PC) – which become large in the close vicinity of $q_T = 0$. As detailed in [1], we count $\epsilon \sim 1/L_\perp \sim \sqrt{\alpha_s}$. In our results, we resum terms up to $\mathcal{O}\left(\epsilon^5\right)$.

The final result for the matched cross-section, as derived in [1], reads

$$f_{\text{matched}} = t\left(q_T\right) f_{\text{res}} + \left[R_{\text{sud}}\left(\mu_{ms}\right)\right]^{t(q_T)} \left[f_{\text{fo}} - t\left(q_T\right) f_{\text{sqt}}\right] \ , \tag{1}$$

where f corresponds either to the single- or double-differential cross section:

$$f = \frac{\mathrm{d}\sigma}{\mathrm{d}q_T} \qquad \text{or} \qquad f = \frac{\mathrm{d}^2\sigma}{\mathrm{d}q_T \mathrm{d}y} \ . \tag{2}$$

Here: y denotes the pseudo-rapidity of the produced boson. The individual terms in (1) refer to: The FO result f_{fo} without any resummation. Its small-q_T expansion f_{sqt}, thus the leading terms in $\frac{q_T^2}{q^2}$ of f_{fo}. The pure resummed result f_{res}. Furthermore we define the matching-correction f_{mc} to the resummed result and the resummation-correction f_{rc} to the FO result as:

$$f_{\text{mc}} := \left[R_{\text{sud}}\left(\mu_{ms}\right)\right]^{t(q_T)} \left[f_{\text{fo}} - f_{\text{sqt}}\right] \qquad \text{and} \qquad f_{\text{rc}} := t\left(q_T\right)\left[f_{\text{res}} - f_{\text{sqt}}\right] \ . \tag{3}$$

The remaining two functions in (1), the *transition function* $t(q_T)$ and the *sudakov-suppression factor* $R_{\text{sud}}\left(\mu_{ms}\right)$, parametrize our ignorance of the resummed form of higher twist operators [1]. $R_{\text{sud}}\left(\mu_{ms}\right)$ mimicis the *sudakov-suppression* on the unresumed terms of the matching-correction, which becomes important at very small transverse-momenta, while the *transition function* ensures a vanishing resummation-correction at $q_T \sim M$. Their explicit form is given in [1].

CuTe combines the computation of all these components for the different processes (W, Z, DY and Higgs) in one program. It is written in C^{++} and implements the formulas given in [1]. The parton-distribution-functions (PDF) are included via the *Les Houches Accord PDF* (LHAPDF) [2] interface in order to give simple access to different PDF sets. Numerical calculations are done using the *GNU Scientific Library* (gsl) [12] and *Cuba library for multidimensional integration* (Cuba) [3]. Both are incorporated into the CuTe-distribution. CuTe provides a variety of options such as the computation of

scale and PDF uncertainties, cuts, matching and transition scheme choices, precision goals and multi-threading. It also inherits the options available in Cuba and LHAPDF. In the following we discuss how to install and use CuTe in sections 2 and 3, respectively. Many of the available options we discuss in section 4. This will also provide additional details about the theoretical framework. Further details and applications can be found in ref. [1].

2 Installation

2.1 Prerequisites (LHAPDF $\geq$ 6.1.0)

Before installing CuTe the LHAPDF-library ($\geq$ 6.1.0) must be installed.
To find out if it is already installed in common directories, one can try to invoke the executable **lhapdf−config**. The option −−version yields the installed version:

```
lhapdf−config −−version
```

If it is installed in uncommon directories, the user has to provide the location of the mentioned **lhapdf−config** script as detailed in Subsection 2.4.
If it was not yet installed, it can be downloaded from `https://www.hepforge.org/archive/lhapdf/`.
LHAPDF follows the same installation procedure as CuTe (see Subsection 2.2 and 2.3).
Note: LHAPDF itself depends on the *Boost C++ libraries*, which can be found at `https://www.boost.org/`. Its installation procedure is described in the provided Getting Started guide's section 1 and section 5.1.

Installing PDF sets

The LHAPDF-library ships without any PDF sets. These must be installed by using the shell command **lhapdf**.

```
lhapdf install SET_NAME
```

A list of available PDF sets can be found at `https://lhapdf.hepforge.org/pdfsets.html`, or will be displayed by invoking:

```
lhapdf list
```

To update LHAPDFs internal list of PDF sets to the list in the link above please use the option:

```
lhapdf update
```

2.2 Basic installation

CuTe follows the standard GNU installation procedure:

1. Download the latest CuTe-package from `https://CuTe.hepforge.org/`.
2. Extract its content.
3. Change into the extracted directory.
4. Run the *configure* script to create the makefiles.
5. Compile the code.
6. Install the executables, libraries and headers.

```
1 wget https://www.hepforge.org/archive/CuTe/CuTe-2.0.0.tar.gz
2 tar xf CuTe-2.0.0.tar.gz
3 cd CuTe-2.0.0
4 ./configure
5 make
6 make install
```

2.3 User defined directories

The installation process will copy the files into the following default directories:

Executables (CuTe, CuTe−config):	BINDIR = PREFIX/bin ,
Libraries (libCuTe.∗, /pkgconfig/CuTe.pc):	LIBDIR = PREFIX/lib ,
Headers (/CuTe/∗.h):	INCLUDEDIR = PREFIX/include ,
with	PREFIX = /usr/local .

This way the installation, depending on your system, often requires super-user privilege, in order to get access to PREFIX:

```
sudo make install
```

To avoid this you can change the defaults by using the following options when invoking the configure script.

−−prefix=PREFIX	−−bindir=BINDIR
−−libdir=LIBDIR	−−includedir=INCLUDEDIR

E.g. to change the PREFIX, and thus all other directories depending on it, to a folder in your home directory:

```
./configure −−prefix=/home/username/CuTe
```

A list of all possible options to configure will be displayed by:

```
./configure −−help
```

2.4 User defined LHAPDF location

CuTe needs the LHAPDF library and PDF locations. If LHAPDF is installed in common directories CuTe extracts these information automatically by invoking the **lhapdf−config** script, which is installed alongside LHAPDF in its BINDIR.
If CuTe cannot find **lhapdf−config** during ./configure, the user has to provide its location by delivering the

LHAPDF_CONFIG_PATH

variable to the configure script.

If e.g. LHAPDF was installed in /home/username/LHAPDF (by invoking the configure script of LHAPDF with −−prefix=/home/username/LHAPDF), the proper command to configure CuTe would be:

```
./configure LHAPDF_CONFIG_PATH=/home/username/LHAPDF/bin
```

3 Usage

In general CuTe expects user written *infiles* which specify the configuration parameters and creates corresponding *outfiles*. The *infiles* should use the name convention NAME.in. The corresponding *outfiles* will be written in the same directory and automatically named as NAME.out.
For example, the command

```
CuTe /home/username/CuTe/example.in
```

will read the *infile* example.in and create an *outfile* example.out in the directory /home/username/CuTe/. Note that this will immediately override a previously existing *outfile* /home/username/CuTe/example.out. In order to handle several similar *infiles* a special option `Basefile` is provided (see Section 3.3).
It is possible to give an arbitrary number of *infiles* as arguments. CuTe will work them all in a row, creating a corresponding *outfile* for each *infile*.
Running CuTe without any argument, will create an example *infile* example.in (Appendix A.1) in the current directory and calculate the corresponding example.out (Appendix A.2).

3.1 Referencing

If you use CuTe please cite one of the publications listed at `https://CuTe.hepforge.org`. If you refer to the program itself please provide in addition the link.

3.2 Infiles

Infiles are textfiles containing case-sensitive "`key = value`" pairs and possibly comments, marked by `#`. Everything after "`key =`" will be interpreted as `value` until a new "`key =`" is found, while newlines, spaces, tabulators and comments are ignored. The order in which the keys appear in the *infile* is irrelevant.
A complete list of all options is given in Section 4 and an example *infile* example.in (Appendix A.1) can be generated by running CuTe without any argument. The latter includes all options in their default values and specifies the allowed parameters in a following comment.

Listing 1 shows a basic *infile* containing all required and the most basic options. It starts with the definition of the initial and final states: Proton-proton-scattering (`collision = PP`) at a centre-of-mass energy (`sqrts`) of 13 TeV and the production of an on-shell higgs-boson (`production = H`) with invariant mass (`M`) of 125 GeV. Note: All mass afflicted parameters throughout the *infiles* are expected in GeV.
It follows a description of the quantities to calculate: The complete matched cross section (`resummation = MATCHED`) as defined in (1) at ϵ^0+N^{3}LL+N^{2}LO accuracy (`order=2` with `power_counting=STANDARD`) and its uncertainty due to scale variation by a factor of $2^{\pm 1}$ (`scale_variation=2`). The scale μ itself is set to our default (`mu = MU*`):

$$\mu_*(q_T) = q_T + q_* \exp \frac{-q_T}{q_*} \,,$$

here the value of q_* is calculated automatically, by solving numerically the following equation:

$$\eta^i(q_*) = 1 \qquad \text{, with} \qquad \eta^i(\mu) = C^i \, \Gamma^{\text{cusp}}_{(0)} \frac{\alpha_s(\mu)}{4\pi} \ln \frac{q^2}{\mu^2} \,. \tag{4}$$

It follows that the value of q_* – and thus μ_* – depends via α_s on the choosen PDF set. For a comparison of different PDFs one might fix its value by hand, by including the `q_star` option into the *infile*.
The chosen PDF set is the N^{2}LO version of the PDF4LHC [13] combination from 2015 (`PDF_set = PDF4LHC15_nnlo_mc`), without uncertainty calculation (`PDF_error`). Note that the chosen PDF set must be installed (see Section 2.1) prior to invoking CuTe.

```
lhapdf update
lhapdf install PDF4LHC15_nnlo_mc
```

The relative accuracy goal of the Cuba integration (`accuracy`) is 10^{-3}, with a maximum number of allowed integrand evaluations (`CUBA_maxeval`) of $5 \cdot 10^6$ per value of q_T and Cuba is free to chose the number of cores by itself (`CUBA_cores<0`).
Finally the transverse-momentum values, CuTe is to calculate the cross section for, are defined. This definition depends on two keys. A list of values `qT` and a modifier `qT_N`. If `qT_N` is set to $n > 1$, CuTe will insert $n - 1$ equally distributed additional values between every consecutive pair in `qT`. It follows that the last two lines in Listing 1 are equivalent to Listing 2 (Note: Values ≤ 0 are ignored).

```
1   collision = PP       #[PP; PPbar; PbarPbar]
2   sqrts = 13000        #[double] GeV
3   production = H       #[DY; Z; W; H]
4   M = 125              #[double] GeV
5   resummation = MATCHED   #[FO; RES; SQT; MATCHED; MC; RC]
6   order = 2            #[0,1,2] N^{1+order}LL / N^{order}LO
7   power_counting = STANDARD  #[NAIVE; STANDARD; IMPROVED]
8   mu = MU*             #[M; QT; MU*]
9   scale_variation = 2  #[double] value>0: mu^{+-} = value^{+-1}*mu
10  PDF_set = PDF4LHC15_nnlo_mc  #[string]
11  PDF_error = 0        #[true; false]
12  accuracy = 0.001     #[double]
13  CUBA_cores = -1      #[int]
14  CUBA_maxeval = 5000000   #[int]
15  qT = 0 8 12 20 60 #[qT1 qT2 ... ] GeV
16  qT_N = 10            #[int]
```

Listing 1: simple.in

```
#      1    2    3    4    5    6    7    8    9    10
qT  =      0.8  1.6  2.4  3.2  4    4.8  5.6  6.4  7.2  # \Delta = 0.8 GeV
       8   8.4  8.8  9.2  9.6  10   10.4 10.8 11.2 11.6 # \Delta = 0.4 GeV
      12  12.8 13.6 14.4 15.2 16   16.8 17.6 18.4 19.2 # \Delta = 0.8 GeV
      20  24   28   32   36   40   44   48   52   56   # \Delta = 4.0 GeV
      60
#qT_N = 1
```

Listing 2: simple_qT.in

3.3 Basefiles

Every *infile* can serve as a *basefile*, meaning that it can be referenced from another infile to determine default options. This is done by the special option `Basefile = path_to_basefile`.
E.g. the following 3 Listings 3-5 use example.in (Appendix A.1) as default and then switch to the pure resummed cross section (`RES`) at the given logarithmic accuracy (`order`).
If only a filename is provided (3), CuTe assumes the *basefile* to be in the same directory as the *infile*, but a relative path (4) from the location of the *infile* or an absolute path (5) is also viable.
As already mentioned, the order of options in *infiles* is irrelevant. If a *basefile* is provided at any place in the *infile*, CuTe will always read the *basefile* first and then overide the results with the options given in the current *infile*. CuTe will e.g. in the case of 5 first read `resummation = MATCHED` from example.in (Appendix A.1) and then overide this option by `resummation = RES`.
A *basefile* can reference another *basefile*, like 4 references 3, which in turn references example.in (Appendix A.1).

```
# Resulting accurracy: e^0 + N^1LL
Basefile    = example.in    # e^0 + N^3LL + N^2LO
resummation = RES
order       = 0
```

Listing 3: example_res_o0.in

```
# Resulting accurracy: e^0 + N^2LL
order       = 1
Basefile    = ./example_res_o0.in    # e^0 + N^1LL
```

Listing 4: example_res_o1.in

```
# Resulting accurracy: e^0 + N^2LL
resummation = RES
Basefile    = /home/dwilhelm/CuTe/example.in   # e^0 + N^3LL + N^2LO
```

Listing 5: example_res_o2.in

3.4 Outfiles

For every given *infile* NAME.in CuTe writes the corresponding results to an *outfile* NAME.out in the same directory. An example *outfile*, the result of example.in (Appendix A.1), is given by example.out (Appendix A.2).
All *outfiles* start with information about the used CuTe version, the starting time of the evaluation, the *infile* name, the complete set of used parameters and a short explanation of the result output. These lines are marked by ## as comments. It follows the table of the calculated results. Its first line, containing the column headers, is market by a single #. The appearing columns are described in Table 1 and 2.

Column Headers	
Header	**Description**
`qT`	Transverse-momentum q_T of the boson in GeV.
`mu`	Resummation scale μ in GeV.
`y`	Pseudo-rapidity y of the boson. Appears only if $\frac{\mathrm{d}^2\sigma}{\mathrm{d}q_T\mathrm{d}y}$ was calculated.
`result`	The final result in pb/GeV.
`RES`, `SQT`, `FO`	The intermediate results in pb/GeV. Appear only if `partial_results=true`.
`D_*+`, `D_*-`	Positive (`+`) and negative (`-`), absolute, signed uncertainties corresponding to `*`. The error-band is given by: $[\texttt{result} + \texttt{D\_*-}, \texttt{result} + \texttt{D\_*+}]$
`D_*`	Absolute, unsigned, symmetric uncertainty corresponding to `*`. The error-band is given by: $[\texttt{result} - \texttt{D\_*}, \texttt{result} + \texttt{D\_*}]$
`D_*+-`	Absolute, signed uncertainty corresponding to `*`. The error-band is given by: $[\texttt{result}, \texttt{result} + \texttt{D\_*+-}]$ for $0 < \texttt{D\_*+-}$, $[\texttt{result} + \texttt{D\_*+-}, \texttt{result}]$ for $\texttt{D\_*+-} < 0$.

Table 1: Column headers of the result table in *outfiles*. The different uncertainty possibilities `*` are given in Table 2.

Error Headers	
Header	**Description**
`D_int`	Numerical integration uncertainty.
`D_mu+`, `D_mu-`	Uncertainty due to scale variation.
`D_PDF`, `D_PDF+`, `D_PDF-`	PDF-uncertainty corresponding to one standard-deviation.
`D_exp+-`	Expansion-scheme uncertainty due to de-/activating `exponent`.
`D_C+`, `D_C-`	Quadratically combined uncertainties.

Table 2: Column headers of the different uncertainties in the result table in *outfiles*.

4 Options

In the following all possible options – except `Basefile` (Subsection 3.3) – for CuTe are explained. Options that are required are marked with a $\star$. For each option there is either given an explicit list of possible values or a type of value. These types are explained in Table 3.
If an option is not specified in the *infile*, its default value is chosen. This value can be altered by the *basefile*.

Value Types	
Type	**Description**
`bool`	Boolean value $\in \{0, 1\}$. `True`/`true` or `False`/`false` are also possible.
`int`	Integer number $\in \mathbb{Z}$.
`double`	Floating point number $\in \mathbb{Q}$.
`string`	Arbitrary sequence of characters.
`type[n]`	A list of exactly `n`, space or tabulator separated values of the type `type`.
`type[]`	A list of space or tabulator separated values of the type `type` with arbitrary length.

Table 3: Different types the value of an option can belong to.

4.1 Process Options

Process Options		
Key	**Value**	**Description**

$\star$ `collision`	`PP`
	`PPbar`
	`PbarPbar`
		Defines the colliding hadrons in the initial state to be either protons `P` or anti-protons `Pbar`.
		Use `PP` for LHC, `PPbar` for Tevatron.

$\star$ `production`	`H`
	`W`
	`Z`
	`DY`
		H: On-shell Higgs production.
		Z: On-shell Z production.
		W: On-shell W^+ production. For W^- use charge conjugation, thus changing `collision` from `PP` to `PbarPbar` and vice-versa.
		DY: Off-shell photon production including the subsequent decay into a lepton pair. Instead of the single-differential cross section $\frac{d\sigma}{dq_T}$ this yields the double-differential cross section $q^2 \frac{d^2\sigma}{dq^2 dq_T}$ in $^{pb}/_{GeV}$.

$\star$ `sqrts`	`double`	The centre-of-mass energy of the collider in GeV.
$\star$ `M`	`double`	The invariant mass of the produced boson in GeV.

4.2 Plot Options

Plot Options			
Key	**Value**	**Default**	**Description**
⋆ `qT`	`double[]`		List of transverse-momentum values q_T in GeV. Values of $q_T \leq 0$ are ignored. If `qT_N` is set to $n > 1$, CuTe will insert $n-1$ equally distributed additional values between every consecutive pair in `qT`, as illustrated in Listings 1 and 2.
`qT_N`	`int`	1	See `qT`.
`y`	`double[]`	Inclusive in y	List of pseudo-rapidity values y. If it is unset, CuTe will calculate the single-differential cross section $\frac{\mathrm{d}\sigma}{\mathrm{d}q_T}$ for every value of q_T in `qT`. Otherwise it will calculate the double-differential cross section $\frac{\mathrm{d}^2\sigma}{\mathrm{d}q_T\mathrm{d}y}$ for every pair (q_T,y).
`y_N`	`int`	1	Modifies the list `y` like `qT_N` modifies `qT`.
`Gridplot`	`int`	0	$\leq \mathbf{0}$ **:** Deactivated. $\geq \mathbf{1}$ **:** For every $Q_T \in$ `q_T`, CuTe calculates the results $f(q_T, \mu) \quad \forall \quad q_T \in X := \{x \in \texttt{q\_T} \vert x \leq Q_T\}$, with $\mu = \mu(Q_T)$. This is needed to generate the *cumulant* of the *spectrum*. If the number of q_T values in X is smaller than `Gridplot`, CuTe inserts additional values.

4.3 Scheme Options

Scheme Options			
Key	**Value**	**Default**	**Description**
⋆ `resummation`	`FO`, `RES`, `SQT`, `MATCHED`, `MC`, `RC`		Choose the different components of the matched cross section to calculate. See equations (1) and (3).
⋆ `order`	int		The perturbative order of the calculation. Depending on `resummation` a value of `order`=n corresponds to $\mathrm{N}^{1+n}\mathrm{LL}$ and/or $\mathrm{N}^n\mathrm{LO}$. Possible values are 0, 1 and 2.

Scheme Options			
Key	**Value**	**Default**	**Description**
⋆ `power_counting`	`NAIVE`, `STANDARD`, `IMPROVED`		Defines the *power-counting* (PC) in the resummed part of the cross-section. `NAIVE` ($\mathrm{N}^{1+n}\mathrm{LL}$): PC in α_s with $L_\lambda \sim \alpha_s^{-1}$. Resums the logarithms of the form L_λ. $n =$ `order`. `IMPROVED` ($\epsilon^e + \mathrm{N}^{1+n}\mathrm{LL}$): PC in ϵ with $\alpha_s \sim \epsilon^2$, $L_\lambda \sim \alpha_s^{-1}$ and $L_\perp \sim \epsilon^{-1}$. Resums in addition the logarithms $L_\perp$ [1]. $e =$ `improved_order` and $n = \lfloor e/2 + 1 \rfloor$. `STANDARD` ($\epsilon^0 + \mathrm{N}^n\mathrm{LL}$): Like `NAIVE`, but un-suppressed terms in the `IMPROVED` PC are correctly resummed (suppressed corresponds to $\mathcal{O}\left(\epsilon^1\right)$).
`improved_order`	`int`	$2 \cdot$ `order` $+1$	Order in ϵ for the resummed part of the cross-section. Only relevant in the `improved` *power-counting*. Should be chosen as $2 \cdot n$ or $2 \cdot n + 1$ for `order` $= n$.
`pi_quadrat`	`bool`	`true`	De-/activate the π^2-resummation in the hard function of the resummed expression. **true:** The hard function is evaluated at $-\mu_h^2$ instead of $+\mu_h^2$, which resums terms proportional to π^2 in the hard function [14]. The size of the not-considered a_s suppressed corrections to the hard function is thus reduced.

Scheme Options			
Key	**Value**	**Default**	**Description**
`exponent`	`bool`	`true`	Expansion scheme for in the choosen PC (`power_counting`) suppressed terms of resummation exponents. **true**: Suppressed terms stay in the exponent. **false**: Suppressed terms are expanded. E.g.: Given a PC in an auxiliary variable a and an arbitrary exponent f, we can expand it as $$f = \sum_{n=-\infty}^{+\infty} a^n f_{(n)} \,.$$ If we furthermore define the square brackets $[\ldots]^m$ as to take only terms up to and including a^m from what is inside $$[f]^m = \sum_{n=-\infty}^{m} a^n f_{(n)} \,,$$ we can write the (un-)suppressed part $f_{(-)+}$ of f as: $$f_- := [f]^0 \quad \text{and} \quad f_+ := f - [f]^0 \,.$$ To approximate the exponential e^f up to a given order $N \geq 0$, neglecting corrections of $\mathcal{O}\left(a^{N+1}\right)$, we can either do the approximation directly in the exponent (`exponent=true`) $$\mathrm{e}^f \approx \mathrm{e}^{[f]^N} = \mathrm{e}^{f_-} \cdot \mathrm{e}^{[f_+]^N} \,,$$ or after expanding the suppressed terms via the series expansion of the exponential (`exponent=false`) $$\mathrm{e}^f \approx \mathrm{e}^{f_-} \cdot \left[\sum_{n=0}^{\infty} \frac{f_+^n}{n!}\right]^N \,.$$ If the PC is correct, the difference between both schemes is of $\mathcal{O}\left(a^{N+1}\right)$ and thus small. It is therefore a measure for the choosen PC.
`exponent_error`	`bool`	`false`	The impact of using the opposite of `exponent` is calculated and presented in the *outfile* as an additional uncertainty.
`matching_scheme`	`NAIVE`, `QT`, `MU`	`QT`	Defines the scale μ_{ms}, for the sudakov-suppression factor R_{sud} in front of the matching-correction (1), as μ_h, q_T or μ. μ_h corresponds to the naive matching-scheme with $R_{\text{sud}} \equiv 1$.

Scheme Options			
Key	**Value**	**Default**	**Description**
`partial_results`	`bool`	`false`	Only affects the composite results `MATCHED`, `MC` and `RC`. If active, CuTe will integrate the individual parts on their own and present the results together with the composite one in the *outfile*. Takes longer and can lead to an unreliable integration uncertainty.
`transition_lrR`	`double[3]`	`1 0 0.01`	Defines the parameter l, r and R (for *left*, *right* and *Reduction*) for the transition-function t in (1). The explicit form depends on `transition_type` and `transition_arg`, but the underlying sigmoid function s fullfills: $$s(l) = 1 - R \quad \text{and} \quad s(r) = R\,,$$ expecting that $0 < R \ll 1$. For more details please refer to the appendix B.
`transition_type`	`NO`, `LOG`, `ERF`, `GPZ`, `MGPZ`	`LOG`	Defines the type of sigmoid function s used for the transition-function t in equation (1). `NO`: No transition-function is used ($t \equiv 1$). `LOG`: The *logistic function*. `ERF`: The *error-function*. `GPZ`: The *Gompertz curve*. `MGPZ`: The mirrored *Gompertz curve*. For more details please see the appendix B.
`transition_arg`	`ETA`, `LAMBDA2`, `LAMBDA`	`ETA`	Defines the argument $x(q_T)$ the transition-function t depends on: `ETA`: $x(q_T) := \frac{\eta^i(q_T)}{\alpha_s(qT)}$. `LAMBDA2`: $x(q_T) := \lambda^2 = \frac{q_T^2}{q^2}$. `LAMBDA`: $x(q_T) := \lambda$. For more details please see the appendix B.

4.4 Scale Options

Scale Options			
Key	**Value**	**Default**	**Description**
`mu`	`M`, `QT`, `MU*`	`MU*`	Sets the scale μ to either M, q_T or our recommended standard $$\mu_*\left(q_T\right) = q_T + q_* \exp\frac{-q_T}{q_*}\,.$$ Has no effect if `mu_value` is set.
`mu_value`	`double`	use `mu`	Fixes the scale μ in GeV. Overrides `mu` option.

Scale Options			
Key	**Value**	**Default**	**Description**
`q_star`	`double`	ODE	Sets the value of q_* which is only used to calculate μ_* (see `mu`). The default is to numerically solve the equation $$\eta^i(q_*) = 1\,,\ \text{with}\ \eta^i(\mu) = C^i\,\Gamma^{\text{cusp}}_{(0)}\frac{\alpha_s(\mu)}{4\pi}\ln\frac{M^2}{\mu^2}\,.$$ The solution depends on the color-factor C^i (C_A for Higgs and C_F otherwise), the Mass `M` and the α_s-routine and perturbative order of the used PDF (`PDF_set`).
`mu_h`	`double`	`M`	Sets the resummation scale μ_h of the hard function. Affects the resummed part of the cross-section and R_{sud}. Should be of the order of the boson mass `M`.
`mu_t`	`double`	172.5	Sets the resummation scale μ_t of the top-quark loop for gluon-fusion. Affects only the resummed part of the Higgs cross-section. Should be of the order of the top-quark mass `M_t`. Default value according to the on-shell mass in [15].
`scale_variation`	`double`	`2.`	A value of `scale_variation` $= f > 0$ activates the uncertainty calculation corresponding to scale variation. In addition to the centre value at μ, the cross-section is evaluated at $\mu^{\pm} = f^{\pm 1}\cdot\mu$. A parabola is fitted into the 3 results and the maximum and minimum values inside $[\mu^-, \mu^+]$ are extracted and presented as scale uncertainties in the *outfile*.
`NPE_model`	`NO`, `GAUSS`, `DIPOLE`, `ENHANCED`	`ENHANCED`	If `Lambda_NP` > 0, this defines the model of *non-perturbative effects* (NPE). `NO` deactivates them, `GAUSS`/`DIPOLE` assumes a simple gaussian/dipole function and `ENHANCED` implements the leading NPE following the derivation in [16]. Affects only the resummed part of the cross section.
`Lambda_NP`	`double`	0.	Sets Λ_{NP} in GeV for NPE. Is expected to be of the order of $\Lambda_{\text{QCD}} \sim 0.2\,\text{GeV}$.

4.5 SM Options

SM Options			
Key	**Value**	**Default**	**Description**
`ckm`	`double[9]`	See [17].	Sets the absolute values of the CKM-Matrix entries, which enter the W production. The order of the 9 numbers is given by: $\lvert V_{ud}\rvert\ \lvert V_{us}\rvert\ \lvert V_{ub}\rvert\ \ \lvert V_{cd}\rvert\ \ldots \lvert V_{tb}\rvert$
`alpha_EM`	`double`	1/127.916	The electromagnetic *fine-structure constant* α_{em}, entering DY, W and Z production.
`G_Fermi`	`double`	1.1663787e-5	The *Fermi constant* G_F, entering Higgs production. Default according to [15].

SM Options			
Key	**Value**	**Default**	**Description**
`theta_W`	`double`	0.49169	Value of the *Weinberg angle* θ_W entering W and Z production.
`M_t`	`double`	172.5	The top-quark mass m_t entering Higgs production. Default value according to the on-shell mass in [15].
`G_cusp_3_factor`	`double`	1	Prefactor f_Γ in front of the padé-approximation to the N^{3}LO *cusp anomalous dimension* $F^i_{(3,0)}$, which enters the `order` ≥ 2 and `improved_order` ≥ 4 resummed contributions: $$\Gamma^i_{(3)} = f_\Gamma \cdot \Gamma^{i,\text{padé}}_{(3)} \, .$$
`F30_factor`	`double`	0	Prefactor f_F in front of the N^{3}LO *anomaly exponent* $F^i_{(3,0)}$, approximated by (4π) times its N^{2}LO coefficient, which enters the `order` ≥ 2 and `improved_order` ≥ 4 resummed contributions: $$F^i_{(3,0)} = f_F \cdot (4\pi) \cdot F^i_{(2,0)} \, .$$
`active_flavour_charges`	`bool[6]`	1 1 1 1 1 1	De-/activates the charge corresponding to a flavour i entering the production currents $ij \to B$. E.g. deactivating the bottom charge sets $b\bar{b} \to Z/\gamma$, $b\bar{j} \to W^-$ and $i\bar{b} \to W^+$ to zero. The order is given by $d\ u\ s\ c\ b\ t$. Note: Quarks whose charge is deactivated still contribute via splitting.

4.6 PDF Options

PDF Options			
Key	**Value**	**Default**	**Description**
⋆ `PDF_set`	`string`		Name of the PDF set to use. Also determines the used α_s-routine and its perturbative order. A list of available sets can be found at `https://lhapdf.hepforge.org/pdfsets.html` Note that the LHAPDF library ships without any PDF sets. These must be installed by using the shell command **lhapdf** (see Section 2.1).
`PDF_error`	`bool`	`false`	De-/activates the PDF-uncertainty calculation. If active, CuTe calculates the cross section with every member of the PDF set. The uncertainty calculation itself is done by LHAPDF, providing that the chosen PDF set supports it.

PDF Options			
Key	**Value**	**Default**	**Description**
PDF_member	int	0	Defines the member of the PDF set to use. Usually the first (PDF_member = 0) member corresponds to the centre value, while other members are used for the PDF-uncertainty.
PDF_dummy	int[2]	use PDF_set	PDF_dummy = a b redefines every PDF to $\phi(z) = z^a(1-z)^b$.
LHAPDF_verbosity	int	0	Sets the verbosity level of LHAPDF.

4.7 Cuba Options

The following options are all inherited from Cuba. We will only shortly explain the most relevant options. For more details please refer to the Cuba manual.

Cuba Options				
	Key	**Value**	**Default**	**Description**
⋆	accuracy	double		Sets the relative accuracy goal the integration should achieve. We recommend a value $\leq 10^{-3}$.
	CUBA_routine	VEGAS, SUAVE, CUHRE, DIVONNE	CUHRE	Defines the integration algorithm. We recommend to use CUHRE.
	CUBA_cores	int	-1	The number of threads used by the Cuba integration. The default and any value < 0 lets Cuba use the approximate number of idle cores. We recommend not to use more threads than cores available (true cores, not virtual ones due to hyperthreading).
⋆	CUBA_maxeval	int		The maximum number of integrand evaluations Cuba is allowed to use for every qT point, to achieve its accuracy goal (accuracy). We recommend a value $\geq 10^6$ (depending on the choosen accuracy goal). For very small values of q_T ($\leq 1\,\text{GeV}$) larger values are necessary.
	CUBA_mineval	int	2000	Defines the minimal number of integrand evaluation Cuba must use for every qT point.
	CUBA_flags	int	0	The first two bits, thus the values 0 to 3, define the Cuba-verbosity level.
	CUBA_seed	int	42	Defines the random-number generator Cuba uses. $= \mathbf{0}$: Sobol (quasi-random). $\neq \mathbf{0}$: Mersenne-Twister (pseudo-random), or Ranlux (pseudo-random), if the bits 8-31 of CUBA_flags are not all 0.

Cuba Options			
Key	**Value**	**Default**	**Description**
CUBA_VEGAS_nstart	int	1000	*Vegas* specific options.
CUBA_VEGAS_nincrease	int	1000	
CUBA_VEGAS_nbatch	int	2000	
CUBA_SUAVE_nnew	int	1000	*Suave* specific options.
CUBA_SUAVE_nmin	int	2	
CUBA_SUAVE_flatness	double	25	
CUBA_DIVONNE_key1	int	47	*Divonne* specific options.
CUBA_DIVONNE_key2	int	1	
CUBA_DIVONNE_key3	int	1	
CUBA_DIVONNE_maxpass	int	5	
CUBA_DIVONNE_border	double	0.001	
CUBA_DIVONNE_maxchisq	double	10	
CUBA_DIVONNE_mindev	double	0.25	
CUBA_CUHRE_key	int	9	*Cuhre* specific option.

4.8 Cut Options

Cut Options			
Key	**Value**	**Default**	**Description**
abs_y_max	double	Inclusive	Restricts the boson pseudo-rapidity to $\|y\| <$ abs_y_max and/or $\|y\| \notin$ abs_y_exclude.
abs_y_exclude	double[2]	Inclusive	
kT_min	double	Inclusive	Assumes the subsequent decay into two massless particles – with 4-momenta k and $\bar{k}$, transverse momenta k_T and $\bar{k}_T$ and rapidities η and $\bar{\eta}$ – and reweights the double differential cross section with the phase space ratio $$r\left(q_T, y\right) = \frac{\Pi\left(q \to k + \bar{k}\right)\big\|_{\text{cut}}}{\Pi\left(q \to k + \bar{k}\right)} \,.$$ The cut in the numerator corresponds, depending on the active options, on a combination of the following 3 restrictions: 1. $k_T >$ kT_min and $\bar{k}_T >$ kT_min. 2. $\|\eta\| <$ abs_eta_max and $\|\bar{\eta}\| <$ abs_eta_max. 3. $\|\eta\| \notin$ abs_eta_exclude and $\|\bar{\eta}\| \notin$ abs_eta_exclude. Note: No branching ratio is multiplied on the production cross section and the angular-dependancy of the cuts is neglected.
abs_eta_max	double	Inclusive	
abs_eta_exclude	double[2]	Inclusive	

A Examples

A.1 Infile Example: **example.in**

```
1 # Infile: example.in
```

```
#  <- comments
#  All mass afflicted parameters in GeV

#--------- Process Options ----------
            collision = PP      #[PP, PPbar, PbarPbar]
                 sqrts = 13000  #[double] GeV
           production = H       #[DY, Z, W, H]
                    M = 125     #[double] GeV

#--------- Scheme Options ----------
          resummation = MATCHED   #[FO, RES, SQT, MATCHED, MC, RC]
                order = 2         #[0,1,2] N^{1+order}LL / N^{order}LO
       power_counting = STANDARD  #[NAIVE, STANDARD, IMPROVED]
#     improved_order = set to 2*order+1   #[2*order or 2*order+1] ~\esilon^{improved_order}
#          pi_quadrat = 1         #[true; false]
#            exponent = 1         #[true; false]
#     exponent_error = 0         #[true; false]
#    matching_scheme = QT        #[NAIVE, QT, MU]
#    partial_results = 0         #[true; false]
#     transition_lrR = 1 0 0.01  #[left,right,R]
#    transition_type = LOG       #[NO, LOG, ERF, GPZ, MGPZ]
#     transition_arg = ETA       #[ETA, LAMBDA, LAMBDA2]

#--------- Scale Options ----------
                  mu = MU*      #[M, QT, MU*]
#            mu_value = use mu #[double] GeV
#              q_star = ode solution #[double] GeV
#                mu_h = M        #[double] GeV
#                mu_t = 172.5    #[double] GeV
     scale_variation = 2        #[double] value>0: mu^{+-} = value^{+-1}*mu
#          NPE_model = ENHANCED  #[NO, GAUSS, DIPOLE, ENHANCED]
#          Lambda_NP = 0         #[double] GeV

#--------- SM Options ----------
#                ckm = 0.97425 0.2253 0.00413 0.225 0.986 0.0411 0.0084 0.04 1.021  #[|Vud| |Vus| |Vub| ... |Vtb|]
#           alpha_EM = 0.00781763   #[double]
#            G_Fermi = 1.16638e-05  #[double]
#            theta_W = 0.491694     #[double]
#                M_t = 172.5        #[double] GeV
#    G_cusp_3_factor = 1            #[double] Gamma_Cusp_3 = Pade * Factor
#         F30_factor = 0            #[double] F_3_0 = F_2_0 * (4 pi) * Factor
#active_flavour_charges = 1 1 1 1 1 1  #[bool d u s c b t]

#--------- PDF Options ----------
              PDF_set = PDF4LHC15_nnlo_mc  #[string]
            PDF_error = 0           #[true; false]
#         PDF_member = 0            #[int]
#          PDF_dummy = use PDF_set  #[int a,int b]
#   LHAPDF_verbosity = 0            #[int]

#--------- Cuba Options ----------
```

```
53                accuracy = 0.001   #[double]
54 #         CUBA_routine = CUHRE   #[VEGAS, SUAVE, CUHRE, DIVONNE]
55              CUBA_cores = -1      #[int]
56           CUBA_maxeval = 5000000    #[int]
57 #            CUBA_flags = 0       #[int]
58 #             CUBA_seed = 42      #[int]
59 #         CUBA_mineval = 2000     #[int]
60 #   CUBA_VEGAS_nstart = 1000      #[int]
61 #CUBA_VEGAS_nincrease = 1000      #[int]
62 #   CUBA_VEGAS_nbatch = 2000      #[int]
63 #     CUBA_SUAVE_nnew = 1000      #[int]
64 #     CUBA_SUAVE_nmin = 2         #[int]
65 #CUBA_SUAVE_flatness = 25        #[double]
66 #   CUBA_DIVONNE_key1 = 47        #[int]
67 #   CUBA_DIVONNE_key2 = 1         #[int]
68 #   CUBA_DIVONNE_key3 = 1         #[int]
69 #CUBA_DIVONNE_maxpass = 5        #[int]
70 #CUBA_DIVONNE_border = 0.001     #[double]
71 #CUBA_DIVONNE_maxchisq = 10         #[double]
72 #CUBA_DIVONNE_mindev = 0.25      #[double]
73 #      CUBA_CUHRE_key = 9         #[int]
74
75 #---------- Cut Options ----------
76 #            abs_y_max = No Cut #[double]
77 #        abs_y_exclude = No Cut #[Min Max]
78 #                kT_min = No Cut #[double]
79 #          abs_eta_max = No Cut #[double]
80 #      abs_eta_exclude = No Cut #[Min Max]
81
82 #---------- Plot Options ----------
83                     qT = 0 8 12 20 60 #[qT1 qT2 ... ] GeV
84                   qT_N = 10       #[int]
85 #                    y = Integrate over y    #[y1 y2 ... ]
86 #                  y_N = 1         #[int]
87 #              Gridplot = 0        #[0=false, N = min evaluations]
```

Listing 6: example.in

A.2 Outfile Example: **example.out**

```
1  ##################################################################
2  ##
3  ##  CuTe 2.0.0
4  ##  Tue Aug  9 20:39:45 2016
5  ##
6  ##################################################################
7  ##
8  ##  Infile: example.in
9  ##
10 ##################################################################
11 ## #---------- Process Options ----------
12 ##              collision = PP      #[PP, PPbar, PbarPbar]
13 ##                  sqrts = 13000   #[double] GeV
```

```
##            production = H       #[DY, Z, W, H]
##                     M = 125     #[double] GeV
##
## #---------- Scheme Options ----------
##           resummation = MATCHED   #[FO, RES, SQT, MATCHED, MC, RC]
##                 order = 2       #[0,1,2] N^{1+order}LL / N^{order}LO
##       power_counting = STANDARD  #[NAIVE, STANDARD, IMPROVED]
## #     improved_order = set to 2*order+1   #[2*order or 2*order+1] ~\
    esilon^{improved_order}
##           pi_quadrat = 1       #[true; false]
##             exponent = 1       #[true; false]
##       exponent_error = 0       #[true; false]
##      matching_scheme = QT      #[NAIVE, QT, MU]
##      partial_results = 0       #[true; false]
##       transition_lrR = 1 0 0.01  #[left,right,R]
##      transition_type = LOG     #[NO, LOG, ERF, GPZ, MGPZ]
##       transition_arg = ETA     #[ETA, LAMBDA, LAMBDA2]
##
## #---------- Scale Options ----------
##                   mu = MU*     #[M, QT, MU*]
## #           mu_value = use mu #[double] GeV
## #             q_star = ode solution #[double] GeV
## #               mu_h = M       #[double] GeV
##                 mu_t = 172.5   #[double] GeV
##      scale_variation = 2       #[double] value>0: mu^{+-} = value^{+-1}*
    mu
##            NPE_model = ENHANCED  #[NO, GAUSS, DIPOLE, ENHANCED]
##            Lambda_NP = 0       #[double] GeV
##
## #---------- SM Options ----------
##                 ckm = 0.97425 0.2253 0.00413 0.225 0.986 0.0411 0.0084
    0.04 1.021  #[|Vud| |Vus| |Vub| ... |Vtb|]
##              alpha_EM = 0.00781763   #[double]
##               G_Fermi = 1.16638e-05  #[double]
##               theta_W = 0.491694  #[double]
##                   M_t = 172.5   #[double] GeV
##      G_cusp_3_factor = 1       #[double] Gamma_Cusp_3 = Pade * Factor
##           F30_factor = 0       #[double] F_3_0 = F_2_0 * (4 pi) * Factor
## active_flavour_charges = 1 1 1 1 1 1   #[bool d u s c b t]
##
## #---------- PDF Options ----------
##              PDF_set = PDF4LHC15_nnlo_mc  #[string]
##            PDF_error = 0       #[true; false]
##           PDF_member = 0       #[int]
## #          PDF_dummy = use PDF_set  #[int a,int b]
##     LHAPDF_verbosity = 0       #[int]
##
## #---------- Cuba Options ----------
##              accuracy = 0.001   #[double]
##          CUBA_routine = CUHRE   #[VEGAS, SUAVE, CUHRE, DIVONNE]
##            CUBA_cores = -1      #[int]
##          CUBA_maxeval = 5000000   #[int]
##            CUBA_flags = 0       #[int]
```

```
##              CUBA_seed = 42      #[int]
##            CUBA_mineval = 2000   #[int]
##     CUBA_VEGAS_nstart = 1000     #[int]
## CUBA_VEGAS_nincrease = 1000      #[int]
##     CUBA_VEGAS_nbatch = 2000     #[int]
##       CUBA_SUAVE_nnew = 1000     #[int]
##       CUBA_SUAVE_nmin = 2        #[int]
##  CUBA_SUAVE_flatness = 25       #[double]
##     CUBA_DIVONNE_key1 = 47      #[int]
##     CUBA_DIVONNE_key2 = 1       #[int]
##     CUBA_DIVONNE_key3 = 1       #[int]
## CUBA_DIVONNE_maxpass = 5        #[int]
##  CUBA_DIVONNE_border = 0.001    #[double]
## CUBA_DIVONNE_maxchisq = 10      #[double]
##  CUBA_DIVONNE_mindev = 0.25     #[double]
##        CUBA_CUHRE_key = 9        #[int]
##
## #---------- Cut Options ----------
## #            abs_y_max = No Cut #[double]
## #       abs_y_exclude = No Cut #[Min Max]
## #               kT_min = No Cut #[double]
## #          abs_eta_max = No Cut #[double]
## #      abs_eta_exclude = No Cut #[Min Max]
##
## #---------- Plot Options ----------
##                    qT = 0 8 12 20 60 #[qT1 qT2 ... ] GeV
##                  qT_N = 10      #[int]
## #                   y = Integrate over y    #[y1 y2 ... ]
##                  y_N = 1        #[int]
##              Gridplot = 0       #[0=false, N = min evaluations]
##
#################################################################
##  Outfile:
#################################################################
#################################################################
## qT : Transverse Momentum of the  Boson
## mu : Factorization Scale
## result    : dsigma/dqT   (matched)
## D_int : Absolute, symmetric Integration Error
## Scale-errors from varying mu in [mu/mu_factor,mu*mu_factor]. Result + D_mu+-
##    D_mu+ : Positive scale error
##    D_mu- : Negative scale error
## D_C    : Signed quadratically combined errors (=+-sqrt(e1+-^2 + ... + en +-^2))
##    D_C+  : Positive
##    D_C-  : Negative
## alpha_s(91.1876) = 0.118002
## q_star(M) = 7.82618
#################################################################
#        qT          mu      result      D_int      D_mu+      D_mu-       D_C+
        D_C-
```

```
113 |  0.8  7.86571  0.150472 1.90728e-05 0.000421394 -0.0255114
        0.000421825 -0.0255114
114 |  1.6  7.97914  0.304075 4.18417e-05 0.000461709 -0.0208278
        0.000463601 -0.0208278
115 |  2.4  8.15927   0.44861 0.000184662 0.00348265 -0.0185393 0.00348754
        -0.0185402
116 |  3.2  8.39965  0.576122 0.000333654 0.00332565 -0.010591 0.00334235
        -0.0105962
117 |    4   8.6944  0.684404 0.000680829 0.00906629 -0.0138714 0.00909182
        -0.0138881
118 |  4.8  9.03824  0.773938 0.000699692 0.0167941 -0.0153779 0.0168086
        -0.0153938
119 |  5.6  9.42641  0.846435 0.000843423 0.0234385 -0.0167991 0.0234537
        -0.0168202
120 |  6.4   9.8546  0.903827 0.000901074 0.0284602 -0.0201177 0.0284745
        -0.0201378
121 |  7.2  10.3189   0.9479 0.000945179 0.0318025 -0.0245233 0.0318165
        -0.0245415
122 |    8  10.8159  0.980789 0.000980651 0.0339007 -0.0294564 0.0339148
        -0.0294727
123 |  8.4  11.0755  0.993457 0.000993022 0.0344807 -0.0318078  0.034495
        -0.0318233
124 |  8.8  11.3422   1.004 0.00100168 0.0348028 -0.0341898 0.0348172
        -0.0342045
125 |  9.2  11.6156  1.01243 0.0010122 0.0347583 -0.0365563  0.034773
        -0.0365703
126 |  9.6  11.8952   1.0191 0.00101851 0.0345561 -0.0389841 0.0345711
        -0.0389974
127 |   10  12.1808  1.02404 0.00102359 0.0340461 -0.0412528 0.0340615
        -0.0412655
128 | 10.4  12.4722  1.02742 0.00102651 0.0334393 -0.0434586 0.0334551
        -0.0434707
129 | 10.8  12.7689  1.02916 0.00102864 0.0328605 -0.0453738 0.0328766
        -0.0453854
130 | 11.2  13.0708  1.02977 0.00102912 0.0322812 -0.0471879 0.0322976
        -0.0471992
131 | 11.6  13.3776  1.02911 0.0010285 0.0317235 -0.0488725 0.0317402
        -0.0488833
132 |   12   13.689  1.02753 0.00102675 0.0311859 -0.0505295 0.0312028
        -0.0505399
133 | 12.8  14.3249  1.02132 0.00102044 0.0300623 -0.0532749 0.0300797
        -0.0532847
134 | 13.6  14.9767  1.01186 0.00101169 0.0288674 -0.055353 0.0288851
        -0.0553622
135 | 14.4   15.643  0.999628 0.000999115 0.0276125 -0.0568055 0.0276306
        -0.0568143
136 | 15.2  16.3222  0.985276 0.000983224 0.0263251 -0.0577168 0.0263434
        -0.0577251
137 |   16  17.0131  0.969169 0.000968167 0.0250228 -0.058186 0.0250416
        -0.0581941
138 | 16.8  17.7147  0.952063 0.0009511  0.023677 -0.0582726 0.0236961
        -0.0582804
```

```
139        17.6    18.4258   0.933452  0.000916317  0.0223412  -0.0579517  0.0223599
       -0.057959
140        18.4    19.1456   0.914307  0.000902683  0.0209978  -0.0573298  0.0210172
       -0.0573369
141        19.2    19.8731   0.894635  0.000891907  0.0196599  -0.0564341  0.0196801
       -0.0564412
142          20    20.6077   0.874389  0.000864672  0.0183543  -0.0552757  0.0183746
       -0.0552825
143          24    24.3645   0.772062  0.000770407   0.012288  -0.0470305  0.0123122
       -0.0470369
144          28    28.2187    0.67451  0.000673263  0.00740321   -0.03707  0.00743376
       -0.0370762
145          32    32.1312   0.586031  0.000581061  0.00379897  -0.0272141  0.00384315
       -0.0272203
146          36    36.0787   0.507992  0.000503468  0.00146249  -0.0268032  0.00154672
       -0.026808
147          40    40.0472   0.440179  0.000437349  0.000246483  -0.0278723
       0.000502024  -0.0278757
148          44    44.0283   0.381502  0.000366991  3.81242e-05  -0.0278924
       0.000368966  -0.0278948
149          48     48.017   0.331059  0.000308352  0.000802814  -0.0272738
       0.000859994  -0.0272755
150          52    52.0102     0.2877  0.000278387  0.00262474  -0.0262367  0.00263946
       -0.0262382
151          56    56.0061   0.250466  0.000230805  0.00511313  -0.0249603  0.00511833
       -0.0249614
152          60    60.0037   0.218482  0.000194496  0.00692256  -0.0235679  0.00692529
       -0.0235687
153 ##############################################################################
154 ## Time:  375.231  sec.
```

Listing 7: example.out

B Transition Functions

We implemented different sigmoid functions $s(x)$. All parametrized by l, r and R, for *left*, *right* and *Reduction*, in such a way that:

$$s(l) = 1 - R \qquad \text{and} \qquad s(r) = R\,, \tag{5}$$

assuming that $0 < R \ll 1$. By introducing furthermore the *centre* m and *width* w:

$$m := \frac{r+l}{2} \qquad \text{and} \qquad w := \frac{r-l}{2} \tag{6}$$

we can write these functions as follows.

$$\begin{aligned}
\textit{Logistic function}: \quad & s_{\log}^{(l,r,R)}(x) := \left[1 + \exp\left\{C_{\log}\frac{x-m}{w}\right\}\right]^{-1}. \\
\textit{Error function}: \quad & s_{\text{erf}}^{(l,r,R)}(x) := \frac{1}{2}\left[1 - \operatorname{erf}\left\{C_{\text{erf}}\frac{x-m}{w}\right\}\right]. \\
\textit{Gompertz curve}: \quad & s_{\text{gpz}}^{(l,r,R)}(x) := \exp\left\{-C_a \exp\left\{-C_b\frac{x-m}{w}\right\}\right\}. \\
\text{Mirrored } \textit{Gompertz curve}: \quad & s_{\text{mgpz}}^{(l,r,R)}(x) := 1 - \exp\left\{-C_a \exp\left\{+C_b\frac{x-m}{w}\right\}\right\}.
\end{aligned} \tag{7}$$

With the R dependant constants:

$$\begin{aligned} C_{\log} &= \ln \frac{1-R}{R} \,, & C_{\mathrm{erf}} &= \mathrm{erf}^{-1}\left\{1-2R\right\} \,, \\ C_a &= \sqrt{\ln\left(1-R\right)\ln R} \quad \text{and} & C_b &= \ln \frac{-C_a}{\ln R} \,. \end{aligned} \tag{8}$$

Their different shapes, depending on R, are shown in Figure 1.

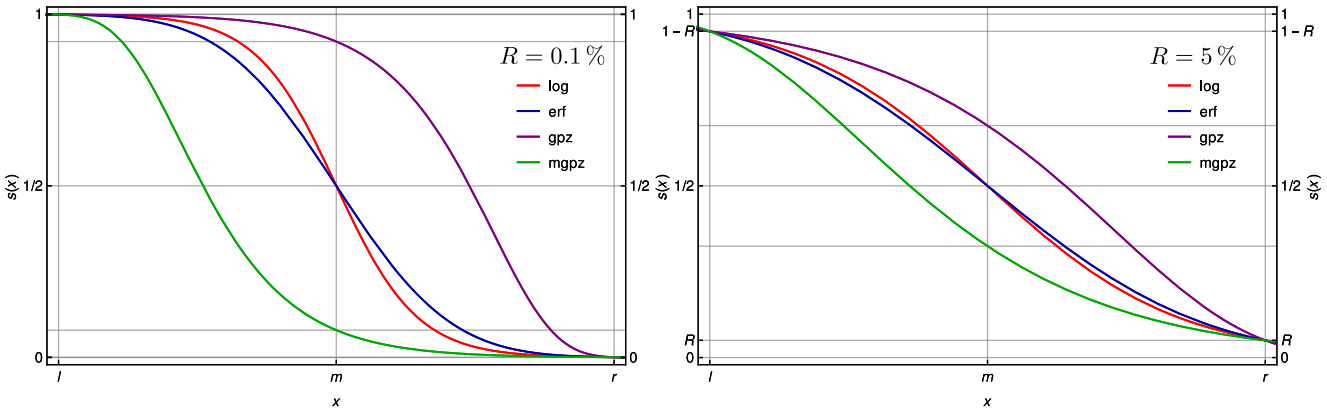

Figure 1: Shapes of the different sigmoid functions (7) for two values of R.

The transition-function $t(q_T)$, which enters the cross section formula (1), is given by the choosen sigmoid function s, using a choosen argument $x(q_T)$ and is normalized to yield 1 at the intercept ($q_T \equiv 0$):

$$t\left(q_T\right) := \frac{s\left(x\left(q_T\right)\right)}{s\left(x\left(0\right)\right)} \,. \tag{9}$$

The implemented arguments are:

$$x(q_T) = \lambda^2 = \frac{q_T^2}{q^2} \,, \qquad x(q_T) = \lambda \qquad \text{or} \qquad x(q_T) = \frac{\eta^i\left(q_T\right)}{\alpha_s\left(q_T\right)} = \frac{C^i \Gamma^{\mathrm{cusp}}_{(0)}}{4\pi} \ln \frac{q^2}{q_T^2} \,. \tag{10}$$

References

[1] T. Becher, T. Luebbert, M. Neubert, and D. Wilhelm, "Precise transverse momentum spectra of massive bosons at hadron colliders," *Work in Progress* (2016) .

[2] A. Buckley, J. Ferrando, S. Lloyd, K. Nordstrom, B. Page, *et al.*, "LHAPDF6: parton density access in the LHC precision era," `arXiv:1412.7420 [hep-ph]`.

[3] T. Hahn, "CUBA: A Library for multidimensional numerical integration," *Comput. Phys. Commun.* **168** (2005) 78–95, `arXiv:hep-ph/0404043 [hep-ph]`.

[4] C. W. Bauer, D. Pirjol, and I. W. Stewart, "Soft-Collinear Factorization in Effective Field Theory," *Phys.Rev.D* **65** (2002) 054022, `hep-ph/0109045`.

[5] T. Becher, A. Broggio, and A. Ferroglia, "Introduction to Soft-Collinear Effective Theory," `arXiv:1410.1892 [hep-ph]`.

[6] T. Becher and M. Neubert, "Drell-Yan production at small q_T, transverse parton distributions and the collinear anomaly," *Eur.Phys.J.* **C71** (2011) 1665, `arXiv:1007.4005 [hep-ph]`.

[7] T. Becher, M. Neubert, and D. Wilhelm, "Electroweak Gauge-Boson Production at Small q_T: Infrared Safety from the Collinear Anomaly," *JHEP* **1202** (2012) 124, arXiv:1109.6027 [hep-ph].

[8] T. Becher, M. Neubert, and D. Wilhelm, "Higgs-Boson Production at Small Transverse Momentum," *JHEP* **1305** (2013) 110, arXiv:1212.2621 [hep-ph].

[9] T. Gehrmann, T. Lubbert, and L. L. Yang, "Transverse parton distribution functions at next-to-next-to-leading order: the quark-to-quark case," *Phys.Rev.Lett.* **109** (2012) 242003, arXiv:1209.0682 [hep-ph].

[10] T. Gehrmann, T. Luebbert, and L. L. Yang, "Transverse Parton Distribution Functions at Next-To-Next-To-Leading-Order," arXiv:1401.1222 [hep-ph].

[11] T. Gehrmann, T. Luebbert, and L. L. Yang, "Calculation of the transverse parton distribution functions at next-to-next-to-leading order," *JHEP* **1406** (2014) 155, arXiv:1403.6451 [hep-ph].

[12] B. Gough, *GNU Scientific Library Reference Manual - Third Edition*. Network Theory Ltd., 3rd ed., 2009. http://www.gnu.org/software/gsl/manual/html_node/.

[13] J. Butterworth *et al.*, "PDF4LHC recommendations for LHC Run II," arXiv:1510.03865 [hep-ph].

[14] V. Ahrens, T. Becher, M. Neubert, and L. L. Yang, "Renormalization-Group Improved Prediction for Higgs Production at Hadron Colliders," *Eur.Phys.J.* **C62** (2009) 333–353, arXiv:0809.4283 [hep-ph].

[15] A. Denner, S. Dittmaier, M. Grazzini, R. V. Harlander, R. S. Thorne, M. Spira, and M. Steinhauser, "Standard Model input parameters for Higgs physics,". https://cds.cern.ch/record/2047636.

[16] T. Becher and G. Bell, "Enhanced non-perturbative effects through the collinear anomaly," *Phys.Rev.Lett.* **112** (2014) 182002, arXiv:1312.5327 [hep-ph].

[17] **Particle Data Group** Collaboration, K. A. Olive *et al.*, "Review of Particle Physics," *Chin. Phys.* **C38** (2014) 090001.