CERN-TH-2025-012, IPPP/25/09, ZU-TH 11/25

NNLOJET: a parton-level event generator for jet cross sections at NNLO QCD accuracy



NNLOJET Collaboration

A. Huss[1,⋆], **L. Bonino**[2], **O. Braun-White**[3], **S. Caletti**[4], **X. Chen**[5], **J. Cruz–Martinez**[1], **J. Currie**[3], **Y.S. Dai**[5], **W. Feng**[2], **G. Fontana**[2], **E. Fox**[3], **R. Gauld**[6], **A. Gehrmann–De Ridder**[2,4], **T. Gehrmann**[2], **E.W.N. Glover**[3], **M. Höfer**[7], **P. Jakubčík**[2], **M. Jaquier**[8], **M. Löchner**[2], **F. Lorkowski**[2], **I. Majer**[4], **M. Marcoli**[3], **P. Meinzinger**[2], **F. Merlotti**[4], **J. Mo**[2], **T. Morgan**[3], **J. Niehues**[3,9], **J. Pires**[10,11], **C. T. Preuss**[12,13], **A. Rodriguez Garcia**[4], **K. Schönwald**[2], **R. Schürmann**[2], **V. Sotnikov**[2], **G. Stagnitto**[14], **H.T. Sun**[5], **D. Walker**[3], **S. Wells**[3], **J. Whitehead**[15], **T.Z. Yang**[2] **and H. Zhang**[16]

1 Theoretical Physics Department, CERN, 1211 Geneva 23, Switzerland
2 Physik-Institut, Universität Zürich, Winterthurerstrasse 190, 8057 Zürich, Switzerland
3 Institute for Particle Physics Phenomenology, Department of Physics, University of Durham, Durham, DH1 3LE, UK
4 Institute for Theoretical Physics, ETH, CH-8093 Zürich, Switzerland
5 School of Physics, Shandong University, Jinan, Shandong 250100, China
6 Max-Planck-Institut für Physik, Werner-Heisenberg-Institut, 85748 Garching, Germany
7 Institute for Theoretical Physics, KIT, 76131 Karlsruhe, Germany
8 Institute for Theoretical Particle Physics, KIT, 76131 Karlsruhe, Germany
9 EKFZ for Digital Health, TU Dresden, 01307 Dresden, Germany
10 LIP, Avenida Professor Gama Pinto 2, 1649-003 Lisboa, Portugal
11 Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal
12 Department of Physics, University of Wuppertal, 42119 Wuppertal, Germany
13 Institut für Theoretische Physik, Georg-August-Universität Göttingen, 37077 Göttingen, Germany
14 Università degli Studi di Milano-Bicocca and INFN, Piazza della Scienza 3, 20126 Milano, Italy
15 Jagiellonian University, ul. prof. Stanislawa Łojasiewicza 11, 31-342 Kraków, Poland
16 PSI Center for Neutron and Muon Sciences, 5232 Villigen PSI, Switzerland

⋆ alexander.huss@cern.ch

Abstract

The antenna subtraction method for NNLO QCD calculations is implemented in the NNLOJET parton-level event generator code to compute jet cross sections and related observables in electron–positron, lepton–hadron and hadron–hadron collisions. We describe the open-source NNLOJET code and its usage.

Copyright attribution to authors.
This work is a submission to SciPost Physics Codebases.
License information to appear upon publication.
Publication information to appear upon publication.

Received Date
Accepted Date
Published Date

Contents

1 Introduction **3**

2 Antenna subtraction **4**

3 The NNLOJET framework **6**
3.1 External dependencies 7
3.2 Installation 7
3.3 Usage of NNLOJET 8

4 Processes in NNLOJET **8**
4.1 Jet production in electron–positron collisions 9
4.2 Jet production in electron–proton collisions 10
4.3 Jet production in hadronic collisions 10
4.4 Vector-boson-plus-jet production 10
4.5 Photon-plus-jet and di-photon production 10
4.6 Higgs-boson-plus-jet production 11

5 Runcard syntax **11**
5.1 General syntax 11
5.2 Observables 12
5.2.1 Reweighting functions 13
5.3 Runcard sections 13
5.3.1 Process block 13
5.3.2 Run block 17
5.3.3 Parameters block 19
5.3.4 Selectors block 20
5.3.5 Histograms block 21
5.3.6 Scales block 24
5.3.7 Channels block 24
5.3.8 Single-line Optional Settings 25

6 Using NNLOJET **26**
6.1 Job submission workflow 26
6.1.1 `init` 26
6.1.2 `submit` 27
6.1.3 `finalize` 28
6.1.4 `config` 30
6.1.5 Run directory 31
6.2 Direct execution 31
6.3 Histogram data files 31

7 Conclusion **32**

A Example: $Z + 1j$ at NNLO **33**

References **35**

1 Introduction

The Standard Model of particle physics is firmly established as the theoretical framework describing the interaction of elementary particles up to the highest energy scales that are accessible at particle collider experiments. Increasingly accurate experimental measurements allow to test the Standard Model and to determine its parameters at an unprecedented level of quantitative precision. To enable these precision physics studies, a close interplay between experimental measurements and theoretical predictions is required.

Many precision observables are derived from hadronic final states, typically in the form of identified jets, which can be related to the production of quarks and gluons (partons) in hard scattering processes [1, 2]. These jet observables provide clean probes of the dynamics of the strong interaction (quantum chromodynamics, QCD) at electron–positron colliders and they affect all types of precision measurements at lepton–hadron or hadron–hadron colliders due to the partonic structure of the colliding hadrons. The theoretical description of these observables must take account of the detailed definition of the final state that is applied in the experimental measurement by incorporating the jet clustering algorithm and any fiducial cuts and isolation requirements on the final-state objects. The resulting predictions for fiducial cross sections can then be compared directly to the experimental data, thereby they stress-test the Standard Model and refine the determination of the parameters (Standard Model couplings and masses, parton distribution functions) that enter the theoretical predictions.

Theoretical predictions rely on a perturbation theory expansion in the coupling constants. In the case of hadron-collider processes and jet observables, QCD corrections are particularly important due to the large value of the strong coupling constant α_s compared to the other Standard Model interactions. At higher orders in perturbation theory, a Born-level process with given final-state multiplicity receives two types of contributions: virtual corrections from closed particle loops in the underlying scattering amplitudes and real-radiation corrections from final states with higher multiplicity. At next-to-leading order (NLO), these correspond to one-loop virtual corrections (V) and single real emission corrections (R). With increasing orders, corrections can be of mixed type, with next-to-next-to-leading order (NNLO) consisting of two-loop double virtual (VV), mixed one-loop single real emission (RV) and tree-level double-real emission corrections (RR) and so on. Each individual subprocess contribution contains infrared singularities arising from soft or collinear momentum configurations in loop integrals and real-radiation processes; only the sum of all subprocesses contributing to a well-defined (infrared-safe, [3]) final state is finite and physically meaningful. These infrared singularities are explicit (as poles in the dimensional regularization parameter) only in the virtual loop amplitudes [4–7], while leading to divergent phase-space integrals in the real-radiation processes. To obtain numerically finite predictions from real-radiation corrections, their infrared singularity structure must be made explicit.

For this task, various infrared subtraction methods have been proposed and implemented for NLO calculations [8–10]. When combined with generic procedures for the evaluation of one-loop virtual amplitudes [11–14] for any process, these methods have enabled the full automation of NLO corrections in QCD and the electroweak theory, becoming part of multi-purpose event simulation programs [15–19], which are complemented by the dedicated libraries for NLO processes in MCFM [20] and VBFNLO [21]. These perturbative fixed-order calculations are implemented as parton-level Monte Carlo event generation programs. Such a program contains the infrared-finite remainders of all parton-level subprocesses. These are evaluated over their respective phase spaces of different multiplicities, producing weighted parton-level events with full kinematic information that can be analysed according to the experimental fiducial cross section definition.

Building upon the universal factorization of QCD real-radiation amplitudes in their infrared

limits [22–26], several NNLO subtraction methods have been formulated [27–38]. These methods have enabled the calculation of NNLO QCD corrections to a large number of collider processes [39] with up to three identified final-state objects. Such calculations are performed on a case-by-case basis in the framework of parton-level Monte Carlo event generation, typically requiring high-performance computing resources to yield stable and numerically reliable results. The resulting codes often require dedicated tuning and post-processing for each observable. They are usually not broadly available as open-source. Notable exceptions are EERAD3 [40, 41] for jet production in electron-positron annihilation, the MATRIX [42] and MiNNLOPS [43, 44] codes for processes with colour-neutral or heavy-quark final states, and the NNLO-upgrade of the MCFM code [20, 45] for selected di-boson processes, as well as a range of codes each covering individual processes at NNLO accuracy [29, 35, 46–52].

In this work, we present the open-source release of the NNLOJET parton-level event generator code, which employs the antenna subtraction method [30,53,54] to compute the NNLO QCD corrections to several classes of jet production processes in hadron–hadron, lepton–hadron and electron–positron collisions. In the following, we review the antenna subtraction method in Sect. 2, describe the NNLOJET installation and general usage in Sect. 3 and the currently available processes in Sect. 4. The detailed usage of NNLOJET is documented through its runcard syntax and process workflow in Sections 5–6. The NNLOJET code is available on HEPForge: http://nnlojet.hepforge.org and is distributed under the GNU General Public License v3.0.

2 Antenna subtraction

The differential cross section $\mathrm{d}\sigma$ for a specific final state at a hadron collider is obtained by the convolution of parton distributions $f_{i|h}$ with parton-level cross sections $\mathrm{d}\hat{\sigma}$

$$\mathrm{d}\sigma = \sum_{a,b} \mathrm{d}\hat{\sigma}_{ab} \otimes f_{a|h_1} \otimes f_{b|h_2} \,. \tag{1}$$

These parton-level cross sections are computed in perturbative QCD as an expansion in the strong coupling:

$$\mathrm{d}\hat{\sigma}_{ab} = \mathrm{d}\hat{\sigma}_{ab,\mathrm{LO}} + \mathrm{d}\hat{\sigma}_{ab,\mathrm{NLO}} + \mathrm{d}\hat{\sigma}_{ab,\mathrm{NNLO}} + \dots \,. \tag{2}$$

Their computation must account for the definition of the final state (jet and object reconstruction as well as fiducial cuts on the final-state objects) and allow the extraction of differential distributions in all desired kinematic variables. This flexibility can only be attained by analysing all parton-level contributions to the cross section at a given perturbative order at the level of the respective partonic kinematics. The resulting implementation is a parton-level event generator, which can compute the differential cross sections for any infrared-safe observable that can be derived from a given Born-level process.

The leading-order contribution $\mathrm{d}\hat{\sigma}_{ab,\mathrm{LO}}$ defines this Born-level process for n final-state objects:

$$\mathrm{d}\hat{\sigma}_{ab,\mathrm{LO}} = \int_n \mathrm{d}\hat{\sigma}^B_{ab} \,, \tag{3}$$

where $\mathrm{d}\hat{\sigma}^B_{ab}$ is the Born-level differential cross section including the n-object final-state definition and $\int_n$ indicates the integration over the n-particle final-state phase space. At LO, each final-state object is described by a single particle, such that the event reconstruction amounts

to a simple check that all final-state objects are non-overlapping in phase space and are all within the fiducial acceptance.

The NLO contribution

$$\mathrm{d}\hat{\sigma}_{ab,\mathrm{NLO}} = \int_n \left(\mathrm{d}\hat{\sigma}^V_{ab,\mathrm{NLO}} + \mathrm{d}\hat{\sigma}^{MF}_{ab,\mathrm{NLO}}\right) + \int_{n+1} \mathrm{d}\hat{\sigma}^R_{ab,\mathrm{NLO}} \tag{4}$$

consists of virtual one-loop corrections $\mathrm{d}\hat{\sigma}^V$, real-radiation corrections $\mathrm{d}\hat{\sigma}^R$, and mass-factorization counterterms $\mathrm{d}\hat{\sigma}^{MF}$ associated with unresolved radiation from the incoming partons. All three contributions are separately infrared-divergent and only their sum is infrared-finite and physically well-defined. The singularities are commonly made explicit by working in dimensional regularization in $d = 4 - 2\epsilon$ dimensions.

At NNLO, the cross section consists of double-virtual (two loop and one-loop squared) corrections $\mathrm{d}\hat{\sigma}^{VV}$, mixed real–virtual corrections $\mathrm{d}\hat{\sigma}^{RV}$, and double-real-radiation corrections $\mathrm{d}\hat{\sigma}^{RR}$, as well as mass-factorization terms $\mathrm{d}\hat{\sigma}^{MF,1}$ and $\mathrm{d}\hat{\sigma}^{MF,2}$ associated with single and double initial-state radiation:

$$\begin{aligned}\mathrm{d}\hat{\sigma}_{ab,\mathrm{NNLO}} = &\int_n \left(\mathrm{d}\hat{\sigma}^{VV}_{ab,\mathrm{NNLO}} + \mathrm{d}\hat{\sigma}^{MF,2}_{ab,\mathrm{NNLO}}\right) \\ &+ \int_{n+1} \left(\mathrm{d}\hat{\sigma}^{RV}_{ab,\mathrm{NNLO}} + \mathrm{d}\hat{\sigma}^{MF,1}_{ab,\mathrm{NNLO}}\right) + \int_{n+2} \mathrm{d}\hat{\sigma}^{RR}_{ab,\mathrm{NNLO}} \,.\end{aligned} \tag{5}$$

All five contributions are infrared-divergent; the cancellation of infrared singularities is accomplished only if they are all summed together.

The infrared singularities at NLO and NNLO are of two forms: explicit ϵ-poles from loop integrals in the virtual corrections [4] and implicit soft and/or collinear singularities in the real corrections [22–26], which become explicit only upon integration over the respective phase space. Consequently, it is not possible to implement equations (4) and (5) directly in a numerical code. To accomplish the cancellation of infrared singularities among the different contributions at NLO and NNLO, one introduces a subtraction procedure that extracts the implicit infrared singularities from real-radiation processes, makes them explicit by phase-space integration and combines them with the virtual corrections. The resulting expressions read

$$\begin{aligned}\mathrm{d}\hat{\sigma}_{ab,\mathrm{NLO}} = &\int_{n+1} \left(\mathrm{d}\hat{\sigma}^R_{ab,\mathrm{NLO}} - \mathrm{d}\hat{\sigma}^S_{ab,\mathrm{NLO}}\right) \\ &+ \int_n \left(\mathrm{d}\hat{\sigma}^V_{ab,\mathrm{NLO}} - \mathrm{d}\hat{\sigma}^T_{ab,\mathrm{NLO}}\right)\end{aligned} \tag{6}$$

and

$$\begin{aligned}\mathrm{d}\hat{\sigma}_{ab,\mathrm{NNLO}} = &\int_{n+2} \left(\mathrm{d}\hat{\sigma}^{RR}_{ab,\mathrm{NNLO}} - \mathrm{d}\hat{\sigma}^S_{ab,\mathrm{NNLO}}\right) \\ &+ \int_{n+1} \left(\mathrm{d}\hat{\sigma}^{RV}_{ab,\mathrm{NNLO}} - \mathrm{d}\hat{\sigma}^T_{ab,\mathrm{NNLO}}\right) \\ &+ \int_n \left(\mathrm{d}\hat{\sigma}^{VV}_{ab,\mathrm{NNLO}} - \mathrm{d}\hat{\sigma}^U_{ab,\mathrm{NNLO}}\right) \,.\end{aligned} \tag{7}$$

The subtraction terms relate among each other as follows:

$$\begin{aligned}
\mathrm{d}\hat{\sigma}^{T}_{ab,\mathrm{NLO}} &= -\int_1 \mathrm{d}\hat{\sigma}^{S}_{ab,\mathrm{NLO}} - \mathrm{d}\hat{\sigma}^{MF}_{ab,\mathrm{NLO}}\,, \\
\mathrm{d}\hat{\sigma}^{S}_{ab,\mathrm{NNLO}} &= \mathrm{d}\hat{\sigma}^{S,1}_{ab,\mathrm{NNLO}} + \mathrm{d}\hat{\sigma}^{S,2}_{ab,\mathrm{NNLO}}\,, \\
\mathrm{d}\hat{\sigma}^{T}_{ab,\mathrm{NNLO}} &= \mathrm{d}\hat{\sigma}^{T,1}_{ab,\mathrm{NNLO}} - \int_1 \mathrm{d}\hat{\sigma}^{S,1}_{ab,\mathrm{NNLO}} - \mathrm{d}\hat{\sigma}^{MF,1}_{ab,\mathrm{NNLO}}\,, \\
\mathrm{d}\hat{\sigma}^{U}_{ab,\mathrm{NNLO}} &= -\int_1 \mathrm{d}\hat{\sigma}^{T,1}_{ab,\mathrm{NNLO}} - \int_2 \mathrm{d}\hat{\sigma}^{S,2}_{ab,\mathrm{NNLO}} - \mathrm{d}\hat{\sigma}^{MF,2}_{ab,\mathrm{NNLO}}\,.
\end{aligned} \tag{8}$$

Here $\int_i$ denotes the integration over the phase space associated with i unresolved partons and the decomposition into $\mathrm{d}\hat{\sigma}^{S,(1,2)}$ separates single- and double-unresolved configurations. By introducing these subtraction terms, each line in equations (6) and (7) is free of implicit and explicit singularities, thus allowing a numerical implementation in a parton-level event generator.

NNLOJET implements the antenna subtraction method [30,53,54], which derives the subtraction terms starting from colour-ordered decompositions of all real-radiation contributions. The colour-ordering allows identifying each unresolved real-radiation configuration with a pair of hard radiator partons, which emit the unresolved partons. This configuration is then described by an antenna function consisting of the hard radiators and unresolved radiation in between them. Antenna functions were derived at NLO and NNLO for quark–antiquark [55], quark–gluon [56] and gluon–gluon [57] pairs as hard radiators. All quarks are treated as massless. The unintegrated subtraction terms are obtained from products of antenna functions and reduced matrix elements of lower partonic multiplicity, which are summed over all configurations of hard radiators and unresolved partons. The kinematics of the antenna functions and the reduced matrix elements are factorized from each other through a phase-space mapping [53,58], such that the antenna function can be integrated over the antenna phase space of the hard radiator partons and the unresolved radiation while leaving the kinematics of the reduced matrix element unaffected. The integrated antenna functions enter the subtraction terms $\mathrm{d}\hat{\sigma}^{T}$ and $\mathrm{d}\hat{\sigma}^{U}$ where they establish the explicit cancellation of infrared ϵ-poles between real and virtual contributions. They were computed up to NNLO for all different configurations of the hard radiators: final–final [30], initial–final [59] and initial–initial [60–62].

With these ingredients, all terms in equation (8) can be implemented in the antenna subtraction formalism, as described in detail in [54]. For all processes available in NNLOJET, this implementation has been made on a process-by-process basis, usually starting from the implicit infrared singularity structure of the double-real-radiation process $\mathrm{d}\hat{\sigma}^{RR}$. A fully automated construction of the antenna subtraction terms in colour space is currently under development [63,64].

3 The NNLOJET framework

The NNLOJET code is a parton-level event generator that provides the framework for the implementation of jet production processes to NNLO accuracy in QCD, using the antenna subtraction method. It contains the event generator infrastructure (Monte Carlo phase-space integration, event handling and analysis routines) and provides the unintegrated and integrated antenna functions and the phase-space mappings for all kinematic configurations. The NNLOJET phase-space integration is based on optimized parametrizations of the respective subprocess phase spaces (described in detail in [65]), which are then integrated using the adaptive Monte Carlo routine VEGAS [66].

The implementation of processes in the NNLOJET framework requires the availability of the matrix elements for all RR, RV, and VV contributions, as well as the construction of the antenna subtraction terms. The subtraction terms have been validated by verifying their pointwise convergence to the respective subprocess matrix elements in all limits, as described in detail in [63, 67]. Some of the one-loop subprocess matrix elements were taken from the MCFM code [20], in particular for processes involving vector bosons [68]. The two-loop virtual corrections to all processes with four-point kinematics or beyond contain various special functions, in particular harmonic polylogarithms [69] and their generalizations. These are evaluated using HPLOG [70], TDHPL [71] and CHAPLIN [72], which are all included as part of the NNLOJET distribution, with explicit consent of the respective code authors and thus in line with the underlying licensing provisions. The evaluation of parton distributions and of the strong coupling constant is performed using the standard LHAPDF [73] interface.

3.1 External dependencies

The NNLOJET code is mainly written in modern Fortran with some modules, dependencies and driver files written in C++ and Python. Multi-threading capabilities are supported through OpenMP. The NNLOJET installation requires compilers for Fortran/C/C++, a Python 3 interpreter (minimum version 3.10), as well as CMake (minimum version 3.18) to be available on the system. In addition, LHAPDF 6 [73] must be present, ideally with the `lhapdf-config` executable searchable from the `$PATH` of the system. The user is responsible for downloading the necessary PDF sets to be used in the calculation.

3.2 Installation

The source files can be downloaded from https://nnlojet.hepforge.org as a tarball, which unpacks into a directory `nnlojet-X.Y.Z`. The following commands are then used to build NNLOJET:

```
$ cd nnlojet-X.Y.Z
$ mkdir build
$ cd build
$ cmake .. [options]
$ make [-jN]
$ make install
```

The compilation can be run on `N` cores in parallel with the `-jN` optional argument. The installation can be customised, by passing additional `[options]` to the `cmake` command. The syntax is `-D <var>=<value>`. The most relevant options are:

- `CMAKE_INSTALL_PREFIX=/path/to/nnlojet`: target directory where to install NNLOJET (default: `/usr/local`);
- `CMAKE_BUILD_TYPE=Debug|Release|RelWithDebInfo|MinSizeRel`: build option for NNLOJET (default: `Release`);
- `LHAPDF_ROOT_DIR=/path/to/LHAPDF`: CMake will attempt to find the library automatically but the path can be supplied manually;
- `Python3_ROOT_DIR=/path/to/python`: CMake will attempt to find a python installation automatically but the path can be supplied manually;
- `DOKAN=ON|OFF`: optionally skip the installation of the `nnlojet-run` script (default: `ON`);
- `OPENMP=ON|OFF`: enable OpenMP support (default: `OFF`).

After successful completion of the build and installation, the NNLOJET executable will be located at `/path/to/nnlojet/bin/NNLOJET`. It is recommended to add the location of the NNLOJET `bin` directory to the system `$PATH` variable.

For system-dependent settings (e.g. compiler preferences and options) the user should consult the `cmake` documentation.

3.3 Usage of NNLOJET

The recommended running mode of NNLOJET is through the Python workflow described in Section 6, which will launch the `NNLOJET` executable. The process selection, parameter input and output control of NNLOJET are steered through a plain-ASCII runcard file. With this file, NNLOJET can also be launched manually from the command line:

```
$ NNLOJET --run <runcard>
```

The NNLOJET executable further offers the optional command-line flags:

```
--help                 -  display command line options
--iseed                -  override the seed number of the runcard
--imember              -  override the PDF member of the runcard
--listprocs            -  query all available processes
--listobs <process>    -  display list of valid observables
```

NNLOJET uses dynamical libraries to load individual processes and initializes many parts of the computational setup at runtime. As a result, the process to be considered, observable selectors and histograms can be specified directly in the runcard in a flexible way without requiring the re-compilation of the code. Any dimensionful parameter is given in units of [GeV]. Cross sections outputs are in [fb].

Complete examples for runcards are provided together with reference output on the NNLOJET webpage https://nnlojet.hepforge.org/examples.html.

4 Processes in NNLOJET

The NNLOJET distribution under https://nnlojet.hepforge.org will be updated on a regular basis, especially by adding new processes. The list of available processes in the current version can be obtained by

```
$ NNLOJET --listprocs
```

For version `v1.0.0` this yields:

```
$ NNLOJET --listprocs
Available processes up to NNLO accuracy for hadron-hadron colliders:
*    Z      : production of a Z-boson;
*    ZJ     : production of a Z-boson plus a jet;
*    WP     : production of a positively charged W-boson;
*    WPJ    : production of a positively charged W-boson plus a jet;
*    WM     : production of a negatively charged W-boson;
*    WMJ    : production of a negatively charged W-boson plus a jet;
*    H      : production of a H-boson;
*    HJ     : production of a H-boson plus a jet;
*    H2     : production of a H-boson with decay to two colour singlets, in particular:
*      - HTO2P     : decay to two photons;
*    H2J    : production of a H-boson plus a jet with decay to two colour singlets, in particular:
*      - HTO2PJ    : decay to two photons;
*    H3     : production of a H-boson with decay to three colour singlets, in particular:
*      - HTO2L1P   : Dalitz decay to (l+l-) and photon;
*    H3J    : production of a H-boson plus a jet with decay to three colour singlets, in particular:
*      - HTO2L1PJ  : Dalitz decay to (l+l-) and photon;
*    H4     : production of a H-boson with decay to four colour singlets, in particular:
*      - HTO4E     : decay to (l+l-) (l+l-);
*      - HTO2E2MU  : decay to (l1+l1) (l2+l2-);
*      - HTO2L2N   : decay to (l1-nu1b) (l2+nu2);
*    H4J     : production of a H-boson plus a jet with decay to four colour singlets, in particular:
*      - HTO4EJ    : decay to (l+l-) (l+l-);
*      - HTO2E2MUJ : decay to (l1+l1) (l2+l2-);
*      - HTO2L2NJ  : decay to (l1-nu1b) (l2+nu2);
```

```
*     GJ      : production of a photon plus a jet;
*     GG      : production of two photons;
*     JJ      : production of two jets;
Available processes up to NNLO accuracy for electron-positron colliders:
*     eeJJ    : production of two jets;
*     eeJJJ   : production of three jets;
Available processes up to NNLO accuracy for lepton-hadron colliders:
*     epLJ    : production of a jet through neutral-current exchange;
*     epLJJ   : production of two jets through neutral-current exchange;
Available processes up to NNLO accuracy for electron-hadron colliders:
*     epNJ    : production of a jet through charged-current exchange;
*     epNJJ   : production of two jets through charged-current exchange;
Available processes up to NNLO accuracy for positron-hadron colliders:
*     epNbJ   : production of a jet through charged-current exchange;
*     epNbJJ  : production of two jets through charged-current exchange;
```

For each process type available at NNLO QCD, it is implicitly understood that NNLOJET also provides the NLO correction to the process with one additional jet and the LO of the process with two additional jets. For example, since $Z+1j$ is available at NNLO, $Z+2j$ can be computed at NLO and $Z+3j$ at LO. To perform NLO and LO calculations of processes with additional jets in the final state, one can simply require a higher number of resolved jets in the runcard file. For example, the NLO correction to $Z+2j$ can be computed via an NNLO calculation for the ZJ process requiring at least two jets.

The processes included in the current release are described in the following. Details on their NNLOJET implementation and on validation tests are described in detail in the respective publications. In general, the following checks have been applied on all process: (1) point-tests [63, 67] and (2) technical-cut checks for RR and RV contributions; (3) finiteness of the RV and VV contributions.

In the point tests, multiple phase space points are generated near each potentially singular infrared limit, with the proximity to the limit being controlled by a one-dimensional parameter. It is then validated that the ratio of the matrix element and subtraction term approaches unity in each limit and that the distribution of the ratios is progressively concentrated in a tighter interval around unity as the control parameter decreases.

The phase space generation in all processes imposes a lower technical cut on all Mandelstam invariants, which is chosen considerably smaller than the smallest scale that is resolved by the final-state definition. For a properly functioning subtraction at RR and RV level, the deep-infrared region in the vicinity of the technical cut gives a negligible contribution to the cross section, which is checked by its independence on the technical cut.

The pole tests are performed numerically for a large number of bulk phase space points on the RV process, where ϵ-poles of matrix elements, mass-factorization terms and subtraction terms are implemented explicitly. In the VV process, all integrated subtraction terms are combined with the mass-factorization terms and the pole terms extracted [4] from the double-virtual matrix element to obtain analytical pole cancellation and to define the finite remainder that is implemented numerically.

4.1 Jet production in electron–positron collisions

Jet final states in e^+e^- annihilation and related event shapes played a key role in establishing QCD as theory of the strong interaction and their theoretical study has been a major driver of developments in precision calculations in QCD. The $e^+e^- \to 3j$ cross section and related event shapes were the first jet observables to be computed to NNLO in QCD [74], originally implemented in the code EERAD3 [40] which is also based on antenna subtraction. The implementation of $e^+e^- \to 2j$ and $e^+e^- \to 3j$ in NNLOJET is described in [75]. It uses the matrix elements for vector-boson decay into partons [76–82], which account for the incoming lepton kinematics. The commonly used LEP-era event-shape variables and jet algorithms [83] are part of the generic NNLOJET infrastructure.

4.2 Jet production in electron–proton collisions

Jet production in deeply inelastic electron–proton scattering (DIS) allows to probe aspects of the proton structure that are difficult to assess in inclusive DIS and provides important precision tests of QCD. The implementation of ep $\to$ lepton + $2j$ up to NNLO QCD is described in [84, 85], including both neutral-current and charged-current DIS. It is based on crossings [81] of the $e^+e^- \to$ jets matrix elements to DIS kinematics. Jets can be defined in the Breit frame or in the electron–proton collider frame and DIS event shapes [86] are part of the NNLOJET infrastructure.

4.3 Jet production in hadronic collisions

Jet production is the most ubiqutous QCD process at hadron colliders, probing the basic $2 \to 2$ parton scattering reactions. A number of kinematic distributions can be derived from single-inclusive jet production $pp \to j + X$ and from dijet production $pp \to 2j$. Their NNLOJET implementation is described in detail in [67, 87–93]. It is based on the relevant subprocess matrix elements up to NNLO [94–106]. The generic NNLOJET infrastructure allows selecting various different jet algorithms, performing scans over jet-resolution parameters and the setting of renormalization and factorization scales [107] either in a jet-based or event-based manner.

4.4 Vector-boson-plus-jet production

Precision studies of the electroweak interaction are commonly performed through massive vector-boson production, identified through their leptonic decay mode. Further kinematic information can be obtained by considering vector-boson-plus-jet final states. Observables derived from vector-boson and vector-boson-plus-jet production are widely used in precision measurements of electroweak parameters and of the proton structure. The NNLOJET implementation for $pp \to (\gamma^*, Z) + 0j$, $pp \to (\gamma^*, Z) + 1j$ is described in [108–114] and for $pp \to W + 0j$, $pp \to W + 1j$ in [112, 115]. The implementations are based on parton-level matrix elements up to NNLO, obtained as kinematic crossings [82] of the $e^+e^- \to$ jets matrix elements. Decays into lepton pairs and the off-shell vector-boson propagator are included. It is also possible to obtain simplified predictions for on-shell vector-boson production in the narrow-width prescription.

4.5 Photon-plus-jet and di-photon production

The production of isolated photons is a classical collider observable. Its theoretical description must account for the experimental prescription that is used to isolate the photons from hadronic activity in the event. This prescription is typically formulated in terms of an isolation cone around the photon direction, which admits only a limited amount of hadronic energy. All relevant experimental measurements use a fixed-size cone, while theory calculations also consider dynamical isolation cones [116], which gradually reduce the allowed hadronic energy to zero towards the photon direction in the cone centre. Only fixed-size cones admit some amount of collinear photon radiation off QCD partons, thus inducing a dependence on photon fragmentation processes [117]. The NNLOJET implementation of $pp \to \gamma + X$ and $pp \to \gamma + 1j$ to NNLO QCD with a dynamical isolation cone is described in [118]. Its extension to fixed-cone isolation is documented in [119], employing an empirical parametrization of the photon fragmentation functions [120]. Di-photon production $pp \to \gamma\gamma$ is currently implemented to NNLO QCD only for dynamical cone isolation [121]. The matrix elements up to NNLO QCD for both processes are taken from [122–125].

4.6 Higgs-boson-plus-jet production

The dominant Higgs-boson-production process at the LHC is gluon fusion, which is mediated through a virtual top-quark loop. The NNLOJET implementation of $pp \to H + 0j$ and $pp \to H + 1j$ to NNLO QCD is described in [126]. It is based on an effective field theory (EFT) which couples the Higgs-boson to the gluon field strength, resulting from integrating out the top-quark loop in the infinite-mass limit [127, 128]. The relevant matrix elements up to NNLO are based on [129–133]. The Higgs boson is considered to be on-shell and a narrow-width approximation is used. Besides the on-shell Higgs production, the Higgs decay modes to 2γ [134], 4ℓ and $2\ell 2\nu$ [135] as well as to $2\ell + \gamma$ [136] are implemented, allowing arbitrary fiducial cuts on the Higgs decay products. Heavy-quark mass corrections are implemented at LO QCD for $pp \to H + 0j$ and $pp \to H + 1j$, allowing for a multiplicative reweighting of the infinite-mass EFT predictions.

5 Runcard syntax

5.1 General syntax

The general syntax of a runcard file follows these rules:

- Each statement should be written on a *separate* line. Wrapping across multiple lines requires the line continuation symbol `&` at the end of the line and at the beginning of the following line.
- Comments start with an exclamation mark `!`, all characters that follow `!` will be ignored by the parser.
- The parser does not distinguish between upper- and lower-case letters, with the exception of observable names (see below) which are case sensitive but typically defined in lower case.
- Assignments follow the syntax `<var> = <val>[...]` with any white space allowed in between and possibly extra options provided inside square brackets `[...]`.
- Lists are specified as `[<val1>,<val2>,...,<valN>]` and white space will be ignored.
- The types of literals follow Fortran syntax:
  - `BOOL`: boolean variables, `.true.` or `.false.`;
  - `INT`: integers;
  - `DBLE`: double precision variables, e.g. `13.6d3` and `1d-8`;
  - `STR`: string expression made up with any ASCII characters (without any quotes).

The runcard is composed of the following separate blocks:

- `PROCESS` **[required]**: defines and specifies the process;
- `RUN` **[required]**: defines the technical parameters and settings;
- `PARAMETERS`: defines the input settings and parameter values;
- `SELECTORS` **[required]**: defines the event and object selectors for the fiducial cross section;

- HISTOGRAMS: defines output histograms (observable, binning, extra options);
- SCALES **[required]**: specifies the set of renormalization and factorization scales for simultaneous evaluation;
- CHANNELS **[required]**: selects the subprocesses to be activated;

as well as optional single-line settings:

- REWEIGHT: weight to bias the sampling of the integrator;
- PDF_INTERPOLATION: switch on dynamical interpolation for PDFs.

They are described in detail in Sect. 5.3.

5.2 Observables

The event selectors for the fiducial cross section, settings for the histograms and values for the scales are specified in terms of observables. The list of available observables depends on the selected process.

Each observable is implemented in a generic way and is internally assigned a unique identifier. Dimensionful observables are defined in [GeV] units. In order to obtain a list of all available observables for a given process with a short description one simply needs to run `NNLOJET --listobs <process_name>` in the terminal. For example, for Z-plus-jet production, it returns:

```
$ NNLOJET --listobs ZJ
*
*    > jet acceptance cuts <
*
*    jets_pt
*    jets_y
*    jets_abs_y
*    jets_eta
*    jets_abs_eta
*    jets_et
*    jets_min_dr_lj
*
*     > process-specific observables <
*
*    ptlm        --- transverse momentum of l-
*    Elm         --- energy of l-
*    ylm         --- (pseudo-)rapidity of l-
*    abs_ylm     --- |ylm|
*    ptlp        --- transverse momentum of l+
*    Elp         --- energy of l+
*    ylp         --- (pseudo-)rapidity of l+
*    abs_ylp     --- |ylp|
*    ptl1        --- transverse momentum of leading lepton
*    yl1         --- (pseudo-)rapidity of leading lepton
*    abs_yl1     --- |yl1|
...
```

Internally, each observable has an attribute to query its validity for an event it was evaluated for. The validity state is typically concerned with observables that are defined based on objects that can vary in multiplicity, such as jets. For example, the transverse momentum of the sub-leading jet in the event `ptj2` is only valid if the event contains at least two jets

after applying all event and jet selection cuts. For a process that only requires one jet to be resolved, e.g. Z + jet, this observable can thus become invalid. While the observable in this case is technically of one order lower in the perturbative accuracy, it is still possible to evaluate it. It is important to keep in mind the validity state of an observable when defining selectors and histograms, as will be discussed below.

5.2.1 Reweighting functions

Reweighting functions define multiplicative factors that are applied to the event weight within a limited context. They can be used in different parts of the runcard to influence various behaviours of the run or the output, as described in Sects. 5.3.5 and 5.3.8. The syntax for reweighting functions is:

```
<observable> ** <power> * <factor> + <offset>
```

- The <observable> is specified using the unique string identifier;
- The <power>, <factor> and <offset> are DBLE;
- The symbols **, * and + can only appear *once* with any white space ignored;
- The ordering of variables and observables does not matter.

For example, all the following lines constitute valid reweighting functions:

```
2               ! a constant factor
ptz             ! an observable
ptz**2          ! an observable raised to a power
3*ptz           ! a rescaled observable
ptz+1           ! an observable with an offset
3*ptz**2+1      ! any combination of the above
```

5.3 Runcard sections

In the following, the runcard input is illustrated in detail. Mandatory settings are highlighted as "**[required]**".

5.3.1 Process block

The PROCESS block defines the process to be calculated and is specified following the general syntax:

```
PROCESS <process_name>
  <option1> = <value1>[<parameters1>]
  <option2> = <value2>[<parameters2>]
  ...
END_PROCESS
```

The currently supported process names (<process_name>) at NNLO can be obtained with the NNLOJET --listprocs command. As described in Sect. 4, they are:

- Z, ZJ: production of Z-boson (plus up to one jet), including Z-boson decay to two charged leptons;
- Wp, Wm, WpJ, WmJ: production of W^+/W^--boson (plus up to one jet), including W-boson decay to a charged lepton and a neutrino;

- H, HJ: production of H-boson (plus up to one jet), excluding H-boson decay;
- H2, H2J: production of H-boson (plus up to one jet) with decay mode to two colour singlets. Specifically:
  - HTO2P, HTO2PJ with decay mode to a γ pair;
- H3, H3J: production of H-boson (plus up to one jet) with decay mode to three colour singlets. Specifically:
  - HTO2L1P, HTO2L1PJ with Dalitz decay mode to $\ell^+\ell^- + \gamma$;
- H4, H4J: production of H-boson (plus up to one jet) with decay mode to four colour singlets. Specifically:
  - HTO4E, HTO4EJ with decay mode to two light lepton pairs with the same flavour;
  - HTO2E2MU, HTO2E2MUJ with decay mode to two light lepton pairs with different flavour;
  - HTO2L2N, HTO2L2NJ with decay mode to a lepton plus a neutrino pair mediated by W-bosons;
- GJ: production of photon plus one jet;
- GG: production of photon pair;
- JJ: production of two jets;
- eeJJ, eeJJJ: production of up to three jets in electron–position colliders;
- epLJ, epLJJ: production of up to two jets in neutral-current lepton-proton collisions, resulting in lepton-plus-jets final states (no distinction is made between leptons and antileptons);
- epNJ, epNJJ: production of up to two jets in charged-current lepton-proton collisions, resulting in neutrino-plus-jets final states;
- epNbJ, epNbJJ: production of up to two jets in charged-current antilepton-proton collisions, resulting in antineutrino-plus-jets final states.

The process names refer to the underlying Born-level process, which can then be used to compute derived quantities with the same final-state kinematics but without identified jets. For example, ZJ also allows the computation of the transverse momentum distribution of the Z-boson, with the jet requirement being replaced by a non-zero transverse momentum of the Z-boson due to partonic recoil. Likewise, GJ contains inclusive photon production, JJ single-inclusive-jet production, and eeJJJ the three-jet-like event shapes.

The currently supported options and values for the PROCESS block are listed in the following.

A collider choice is specified with:

- collider = STR **[required]**: using pre-defined collider setups:
  - proton–proton collider: PP;
  - proton–antiproton collider: PAP;
  - proton–electron collider (deep-inelastic scattering): PE;
  - electron–positron collider: EPEM;

- beam1 = STR, beam2 = STR: More flexibility is provided by setting the beams individually with the following allowed values: P (p), AP ($\bar{\text{p}}$), EM (e^-), EP (e^+), HE ($^4_2$He), LI ($^7_3$Li), C ($^{12}_6$C), O ($^{16}_8$O), CU ($^{64}_{29}$Cu), AU ($^{197}_{79}$Au), PB ($^{208}_{82}$Pb). Setting beam1 and beam2 will override the collider choice. It is the responsibility of the user to choose a compatible (nuclear) PDF set for each beam in the RUN block described below. The centre-of-mass energy is specified for the nucleon–nucleon system. In case of an asymmetric beams setup (beam1 $\neq$ beam2), NNLOJET will further perform the appropriate longitudinal boost of the hadronic centre-of-mass system. The collider coordinate system is then chosen such that beam1 is in the positive z direction.

The collider energy is provided by setting either one of the following options:

- sqrts = DBLE **[required]**: definition of centre-of-mass energy;
- Ebeam1 = DBLE, Ebeam2 = DBLE: definition of energy of individual beams instead of specifying sqrts, allows for asymmetric collisions. Setting Ebeam1 and Ebeam2 will override the sqrts value.

The standard jet reconstruction algorithms are implemented directly in NNLOJET and can be specified with the following options:

- jet = <algo>[<options>]: specification of jet algorithm; the option in DBLE specifies the jet-resolution parameter. The algorithm can be chosen with the following syntax:
  - anti-k_T algorithm [137]: antikt[DBLE];
  - Cambridge–Aachen algorithm [138] (longitudinally invariant): ca[DBLE];
  - k_T algorithm [139, 140] (longitudinally invariant): kt[DBLE];
  - Durham algorithm [141] (for e^+e^- collisions): durham[DBLE];
  - Jade algorithm [142] (for e^+e^- collisions): jade[DBLE];
  - No algorithm required: none;
- jet_exclusive = BOOL: jet exclusive flag, specifying whether events with more jets than required should be rejected (default: .false.);
- jet_recomb = STR: jet-recombination scheme: V4, to combine jets by adding the 4-vector of the momenta, and ET, to combine jets with E_T as the weights (default: V4).

Several optional settings in the PROCESS block are only relevant for specific classes of processes:

- V_NC = STR: restriction on virtual neutral-current gauge-bosons. Options for STR:
  - ZGAMMA (default): both Z bosons and photons are allowed in the amplitudes;
  - GAMMA: only photons, useful for example in DIS processes;
  - Z: only Z bosons are allowed;
- L_NC = STR: decay mode of Z bosons decay to leptons. Values:
  - NU: decay to neutrinos (single family) $Z \to \nu\bar{\nu}$;
  - L (default): decay to charged leptons (single family) $Z \to \ell^+\ell^-$.
- decay_type = INT: definition of heavy-particle decay types used in the phase-space generator. It is set automatically for each process and unstable particle, but can be overridden using the following INT values:

- 0: use $\mathrm{d}M^2/M^2$;
- 1: use resonance parametrization;
- 2: narrow width inserted, delta function;
- 3: on-shell heavy particle, no decay.

The isolation of photons and leptons is defined using the following options:

- `photon_isolation`/`lepton_isolation`: define the isolation of non-QCD particles, as documented in detail in [119]. It allows the combination of several isolation prescriptions that must be simultaneously fulfilled, using the syntax
  `<isolation>` = `<algo1>[<params1>]` + `<algo2>[<params2>]` + `...` .
  For lepton isolation, only fixed-cone isolation is supported. For photon isolation, the following isolation prescriptions are available:
  - `none`: no isolation required;
  - `fixed`: fixed-cone isolation. The non-QCD particle with transverse momentum p_T is isolated if the hadronic transverse energy in a cone of size R_{cone} does not exceed

$$E_T^{\text{max}} = E_T^{\text{const}} + \epsilon p_T \,. \tag{9}$$

    Parameters:
    - `R_cone`: isolation-cone size R_{cone};
    - `ETmax_const`: E_T^{const};
    - `ETmax_epsilon`: ϵ;
  - `smooth`: smooth-cone isolation [116]. The non-QCD particle with transverse momentum p_T is isolated according to a profile function $\mathcal{X}(R,n)$ that permits a decreasing amount of hadronic energy $E_T^{\text{smooth}} = \mathcal{X}(R,n)\,E_T^{\text{max}}$ within sub-cones of radius $R < R_{\text{cone}}$ and E_T^{max} as defined in (9):

$$\mathcal{X}(R,n) = \left[\frac{1-\cos(R)}{1-\cos(R_{\text{cone}})}\right]^n .$$

  - `hybrid_smooth`: hybrid isolation. A variant of the smooth-cone isolation defined in [143]. The following parameters can be specified:
    - `ETmax_const`: E_T^{const};
    - `ETmax_epsilon`: ϵ;
    - `R_cone`: the smooth-cone size or the outer fixed-cone size in hybrid isolation;
    - `n`: parameter n in profile function;
    - `R_inner`: inner smooth-cone size in case hybrid isolation is used, where `R_inner<R_cone` must be satisfied;
  - `democratic`: democratic isolation. The jet clustering procedure includes the non-QCD object, which is then called isolated if it carries more than a fraction z_{cut} of the transverse energy of the jet it is assigned to. Parameters:
    - `zcut`: threshold of transverse-energy fraction between the photon and the jet. If the transverse energy fraction is larger than `zcut`, the clustered particles are considered as one photon; if the fraction is smaller than `zcut`, the clustered particles are considered as one jet.
- `photon_recomb` = `DBLE`: angular distance R below which two photons are recombined;

- photon_fragmentation = STR: definition of photon fragmentation setup. The parameteter specifies the parametrization of the photon fragmentation functions:
  - ALEPH: a simple parametrization obtained from a one-parameter fit by ALEPH [144];
  - BFG0, BFGPERT: BFG0 [120];
  - BFG1, BFGI: BFGI [120];
  - BFG2, BFGII: BFGII [120].

Finally, several PROCESS settings are relevant only for DIS:

- pol = DBLE: polarization fraction of incoming lepton beam (default: 0.0);
- lab_ordereta = BOOL: a flag to order jets in the laboratory frame in η (default: .false.);
- dis_eventshapes = DBLE: minimum value of hadronic energy in an event for DIS event shapes (default: 0.0);
- eprc = BOOL: switch to use a running electromagnetic coupling evaluated at the scale Q^2 (default: .false.);
- dis_frame = STR: definition of the DIS frame, with the following options:
  - FIXT: fixed target system;
  - BREIT: Breit frame;
  - LAB: the laboratory frame.

  The default is BREIT for epLJJ, epNJJ, epNbJJ and LAB for epLJ, epNJ, epNbJ.

5.3.2 Run block

This block defines technical settings for the NNLOJET execution. Runs in NNLOJET consist of two phases:

- The warmup phase aims to construct VEGAS grids that are adapted to the integrand at hand (event-selection cuts, regions of larger weights etc.). The VEGAS grids are initialized as uniform distributions in the raw random variables, and the calculation proceeds in multiple iterations. After each iteration, the grids are adapted according to the accumulated cross section for importance sampling. The events sampled during the warmup phase *do not* enter the final result. For performance reasons, histogram filling and scales beyond the central scales are deactivated.
- The production phase performs the Monte Carlo integration for the total and differential cross sections according to the VEGAS grid that was adapted during the warmup. The VEGAS grids are frozen at this stage and will no longer be updated.

To run NNLOJET, at least one of the two phases needs to be activated. The production phase can only be performed if a dedicated VEGAS grid was produced by a warmup phase beforehand. When the nnlojet-run workflow is used to run NNLOJET, the warmup and production phases are automatically handled and can be omitted from the runcard.

The general syntax for the RUN block is:

```
RUN <run_name>
  <option1> = <value1>[<parameters1>]
  <option2> = <value2>[<parameters2>]
  ...
END_RUN
```

The field <run_name> specifies the user-defined name for the run and it is used to name the output files. The <run_name> for a production phase must be the same as the respective warmup phase for the VEGAS grids to be properly loaded. The options are:

- PDF = <PDF_set>[<member>] **[required]**: specification of a PDF set according to the naming convention of LHAPDF [73]. For asymmetric colliders, separate PDFs can be specified for each beam by instead providing PDF1 and PDF2 individually. The optional <member> = INT parameter specifies the PDF member. It defaults to 0, i.e. the central member of the PDF set. Note that the PDF is required also for electron-positron collisions, for the running of α_s;

- iseed = INT: specification of the seed for the random number generator (default: 1).;

- warmup = <ncall>[<niter>]: specification of warmup run settings. <ncall> = INT indicates the number of integrand evaluations per iteration and <niter> = INT the number of iterations. Instead of <niter>, it is also possible to specify auto, which will automatically distribute the total number of <ncall> evaluations of the integrand into a staggered batch of iterations (at least one of warmup or production needs to be specified);

- production = <ncall>[<niter>]: specification of production run settings; in particular, <ncall> = INT indicates the number of integrand evaluations per iteration and <niter> = INT the number of iterations (at least one of warmup or production needs to be specified);

- tcut = DBLE: technical cutoff defined relative to the squared partonic centre-of-mass energy $\hat{s}$. Events with any invariant smaller than tcut$\cdot\hat{s}$ are rejected in the phase-space integration. The technical cut must be chosen considerably smaller than the smallest scale that is resolved by the final-state definition (default: 1d-8);

- reset_vegas_grid = BOOL: switch for overwriting the local VEGAS grid; when set to .true., the warmup run will be started with a new uniform VEGAS grid; when set to .false., the warmup run will continue starting from the previous VEGAS grid if one exists (default: .false.).

NNLOJET also provides some optional optimization settings which can speed up the calculation when used properly. They are in general only intended for expert users. The following optimization settings are available:

- angular_average = BOOL: switch to average over azimuthal angle rotations in unresolved regions, required to make antenna subtraction terms strictly local (default: .true.);

- cache_kinematics = BOOL: switch to cache all kinematics of mapped momentum sets (default: .false.);

- scale_coefficients = BOOL: switch to internally use a decomposition of weights into scale coefficients of scale logarithms. Necessary for scale overrides (default: .false.);

- multi_channel = <N>: option for multi-channel adaption over individual channels. For the definition of channels, see 5.3.7. The parameter <N> is of type INT. If <N> is 0, every existing channel is evaluated in each sampled phase-space point. Multi-channeling is activated by selecting a non-zero <N>, leading to a selective evaluation of different channels by distributing events among them. An importance sampling is applied by evaluating those channels more often which have a larger contribution to the cross section. The

Quarks (Up type)	Up U	Charm C	Top T
Quarks (Down type)	Down D	Strange S	Bottom B
Leptons	Electron EL	Muon MU	Tau TAU
Bosons	Higgs H	W-boson W	Z-boson Z
Others	Proton PROTON		

Table 1: Acronyms for particles to be used as <particle> of attributes.

strategy for dividing the channels into groups is determined by the value of <N>. If <N> is positive, channels will be grouped such that every phase-space point will evaluate a subset of <N> channels. If <N> is negative, the total number of groups is specified, i.e. the integrator will attempt to divide active channels into |<N>| equal-sized groups (default 0);

- lips_reduce = BOOL: switch for the reduced phase-space mode, a variant of the phase-space generator that exploits the permutation symmetries of a specific process (default: .false.).

5.3.3 Parameters block

The PARAMETERS block defines the Standard Model input parameters of the calculation. The general syntax follows the pattern:

```
PARAMETERS
  <variable>[<spec>] = <value>
  <option> = <value>
  ...
END_PARAMETERS
```

The attributes of particle <particle> are set with the syntax <attribute>[<particle>]. <particle> may assume any acronym from Table 1 as its value. The possible attributes are listed in the following, with the default value taken from [145]:

- mass[<particle>] = DBLE: the mass of the particle, which can be set for the massive particles in the Standard Model and the proton. The mass values correspond to the mass scheme defined below;

- width[<particle>] = DBLE: the decay width of the particle. Z, W, H and top quark are supported. The widths correspond to the scheme defined below.

Further parameters and prescriptions can be set as follows:

- hard_photon_alpha0 = BOOL: switch to use α_0 as the electroweak coupling for resolved photons in photon-production processes (default: .false.);

- scheme[<scheme>] = STR: the specification of schemes in mass, width and electroweak coupling. The options for <scheme> are:

  - MV: define the mass scheme for electroweak gauge bosons. Possible values are:
    * OS (default): on-shell scheme;
    * POLE: pole scheme;
    * MSBAR: $\overline{\text{MS}}$-scheme;
  - GV: define the width schemes for electroweak gauge bosons. Possible values are:
    * NWA: for the narrow-width approximation;
    * CMS: for complex-mass scheme;
    * NAIVE (default): for naive fixed-width scheme;
    * RUN: for running on-shell scheme;
    * POLE: for pole scheme;
  - ALPHA: the electroweak coupling prescription. Possible values are:
    * GMU (default): G_μ scheme, derived from the Fermi constant and the Z and W masses (override default value with GF = DBLE);
    * FIX: fixed scheme (override default value with alpha = DBLE);
    * ALPHA0: α_0 scheme (override default value with alpha0 = DBLE);
    * ALPHAMZ: $\alpha(M_Z)$ scheme (override default value with alphaMZ = DBLE);

- CKM = STR: the specification of the CKM matrix. Users can change the entries by modifying driver/core/CKM.f90 if necessary. The available options are:
  - IDENTITY (default): diagonal CKM matrix;
  - FULL: Wolfenstein parameters [146];
  - CABIBBO: Cabibbo mixing angle only [146];
  - MCFM: the default setup of MCFM.

5.3.4 Selectors block

In NNLOJET, two types of selection criteria are distinguished: object-level and event-level criteria. Object-level criteria may be used to impose conditions on the reconstructed objects, i.e. jets and photons. For example, they allow to select jets with a minimum transverse momentum or within a specific rapidity range. Any object that does not fulfill these conditions is omitted from the event record.

Event-level criteria define which event candidates should be used to calculate cross sections and fill histograms. If an event-level criterion is not fulfilled, the entire event candidate is discarded. Typical examples are a minimum number of jets per event, additional constraints on the leading jet, or constraints on leptons.

The SELECTORS block specifies the object-level and event-level selection criteria. It is encapsulated by

```
SELECTORS
  ! put selectors here
END_SELECTORS
```

It can be left empty if the process is well-defined (corresponding to a fully-inclusive event selection) or filled with any number of selector statements of the form:

```
<type> <observable> min=<val_min> max=<val_max>
```

- The <type> can be either select or reject. In case partially valid observables are expected, one can use select_if_valid or reject_if_valid. An event is selected only if it passes all select statements and is rejected by none of the reject statements.

- The <observable> is specified using a pre-defined string identifier. Possible values for a given process can be checked using the `--listobs` query of the `NNLOJET` command. Object-level cuts are listed in the `photon cuts` and `jet acceptance cuts` sections, while event-level selectors are listed in the `process-specific observables` section.
- It is only necessary to specify at least one of the two assignments for `min` and `max`. If unspecified, the other boundary defaults to $\pm\infty$. If both are specified, the order of the assignments is irrelevant.

Selector statements can be given in any order. Since each selector will internally generate a new object, it is better to have as few lines as possible. It is therefore preferable to write

```
select ylp min = -5 max = +5   ! --> accept [-5, +5]
```

or

```
select abs_ylp max = +5        ! --> accept |y| < 5 => [-5, +5]
```

instead of

```
select ylp min = -5            ! --> accept [-5, +infinity]
select ylp max = +5            ! --> accept [-infinity, +5]
```

It is possible to define `OR` groups, where all the selector statements inside the group have 'or' relations. The groups are encapsulated by

```
OR
  ! put selectors here
END_OR
```

It is not possible to use `OR` groups on object-level selectors.

A common example use case of the `reject` type in the selectors is the exclusion of the barrel–endcap gap region that can be implemented as follows:

```
select abs_ylp max = +5                      ! global selector
reject abs_ylp min = +1.4442 max = +1.560    ! barrel-endcap
```

which is more intuitive and cleaner than the alternative only using `select` and an `OR` group:

```
OR
  select ylp min = -5       max = -1.560
  select ylp min = -1.4442  max = +1.4442
  select ylp min = +1.560   max = +5
END_OR
```

5.3.5 Histograms block

The `HISTOGRAMS` block specifies all the histograms to be filled during the run. It is encapsulated between the lines

```
HISTOGRAMS
  ! put histograms here
END_HISTOGRAMS
```

It can be populated with histogram statements of the following two types

```
<observable> nbins=<val_nbins> min=<val_min> max=<val_max>
<observable> [edge0,edge1,edge2,...,edgeN]
```

- The first version corresponds to a histogram with uniformly-space (linear or logarithmic) bins, whereas the second version specifies the bin boundaries as a list and allows for flexible bin edges.

- The <observable> is specified using the unique string identifier and events giving rise to an invalid attribute (see Sect. 5.2 for the definition) are skipped in the accumulation.
- As a decorator to the <observable> specification, it is possible to specify a file-name override using the notation <observable> > <dat_name>. If this specification is omitted, the file name will use the observable name by default. This option is needed if there exist multiple histograms that bin the same observable (see example below) in order to avoid file-name clashes.
- For cross sections, the special name `cross` can be used instead of the histogram statements above. NNLOJET will *always* register such a histogram by default so that user-defined cross section statements must always be of the form `cross` > <dat_name> with a file-name override. It should be noted that histograms keep track of overflows such that the cross section can also be calculated from the histogram contents by summing the bins and the overflow. It is therefore rarely necessary to define a cross section histogram explicitly.

Every histogram statement can further be supplemented by optional settings. The available options are:

- `fac` = <rwg_fct>: multiply all weights passed to the histogram with a generic reweighting function following the syntax described in Sect. 5.2.1;
- `mu0` = (`DBLE` *)<observable>: override scale. This options sets the both factorization and renormalization scale to the specified value. Only available if `scale_coefficients` = `.true.` is set in the `RUN` block;
- `binning` = `STR`: switch between linear and logarithmic binning, in case equal-sized bins are used. Possible values are:
  - `LIN` (default): linear binning;
  - `LOG`: logarithmic binning;
- `output_type` = `INT`: specify the luminosity-breakdown type: a value of `0` will only output the numbers for the total (default), while a value of `1` will in addition provide numbers broken down into separate partonic luminosities: (*e.g.* $\bar{q}\bar{q}$, $g\bar{q}$, $q\bar{q}$, $\bar{q}g$, gg, qg, $\bar{q}q$, gq, qq). See Sect. 6.3 for further description of the output data files;
- `cumulant` = `INT`: option for cumulant distributions with possible values:
  - `0` (default): standard differential distribution $\mathrm{d}\sigma/\mathrm{d}$<obs>;
  - `-1`: cumulant distribution $\int_{o_{\min}}^{o_i} \mathrm{d}\sigma/\mathrm{d}$<obs>;
  - `+1`: cumulant distribution $\int_{o_i}^{o_{\max}} \mathrm{d}\sigma/\mathrm{d}$<obs>.

  The values $o_{\min}$ and $o_{\max}$ respectively correspond to the lower and upper bounds of the histogram and the user must ensure that it covers the desired kinematic range. Observable values outside of this range are ignored in the cumulant output and are instead collected in an overflow bin.

A further optional specification is given by histogram selectors which can be assigned right below a histogram definition as follows:

```
<observable> ...
HISTOGRAM_SELECTORS
  ! put selectors here
END_HISTOGRAM_SELECTORS
```

with the selector syntax defined in Sect. 5.3.4 but without the support of OR blocks. This allows to generate multi-differential distributions in a flexible way as follows:

```
ptz > ptz_bin1 nbins=75 min=0 max=150
HISTOGRAM_SELECTORS
  select abs_yz max=0.3
END_HISTOGRAM_SELECTORS

ptz > ptz_bin2 nbins=30 min=0 max=90
HISTOGRAM_SELECTORS
  select abs_yz min=0.3 max=1
END_HISTOGRAM_SELECTORS

ptz > ptz_bin3 [0,10,15,20,25,30,40,50,60,80,100]
HISTOGRAM_SELECTORS
  select abs_yz min=1 max=2
END_HISTOGRAM_SELECTORS
```

Here, the modified file names (> ptz_bin<i>) are needed in order to avoid file name clashes. Furthermore, we can choose different ptz-binnings for each yz bin. The histogram file will correspond *precisely* to what is specified in the runcard, for the example above this means

$$\texttt{ptz_bin1}: \qquad \left.\frac{\mathrm{d}\sigma}{\mathrm{d}p_{\mathrm{T,Z}}}\right|_{|y_{\mathrm{Z}}|<0.3} \neq \frac{\mathrm{d}^2\sigma}{\mathrm{d}p_{\mathrm{T,Z}}\,\mathrm{d}y_{\mathrm{Z}}} \quad \text{(in the first } y_{\mathrm{Z}} \text{ bin)}. \tag{10}$$

In order to obtain the double-differential distribution on the right-hand-side of this expression one has to further divide the histogram contents by the bin-width for $y_{\mathrm{Z}} \in [-0.3, 0.3]$ which is 0.6 in this case.

The way invalid observables (see Sect. 5.2 for the definition) are treated within the histogram selectors deserves special care. Once an observable is used in a select or reject statement within the HISTOGRAM_SELECTORS block, it is *implicitly assumed* that it must be valid for all events that are considered for the histogram filling. This means that if an invalid observable is encountered for such a specification, the weight associated with that event is discarded in the accumulation of the histogram. If this is not the desired behaviour, it is instead possible to use the select_if_valid or reject_if_valid specifications that allow to ignore the selector in case the observable is evaluated with an invalid attribute.

Some observables are defined in a composite manner, i.e. they can potentially induce *multiple* bookings to the same histogram from different observables. For this purpose, NNLOJET defines the concept of a composite histogram that follows the following syntax:

```
COMPOSITE > <dat_name> ... <opt_comp>
  <obs1>  <opt1>
  <obs2>  <opt2>
  ...
END_COMPOSITE
```

where ... at the top can be the fixed-bin or the variable-bin specification and <opt_comp> the optional specifications both described above. The composite histogram will be filled with the sum of all the sub-histograms defined inside the block, where each can specify individual options, i.e. fac and mu0 as described above. Both the composite histogram as well as the sub-histograms can further be supplemented by histogram selectors. In case the observable of the sub-histogram is invalid, it will simply be skipped in the accumulation.

A non-trivial example of this feature is the inclusive-jet distribution in a specific rapidity bin and a jet-based scale setting such as the jet-p_T. In this case, we must register as many sub-histograms as the maximum possible number of jets (invalid observables will be skipped),

each dressed with a selector block and further ensure that for each sub-histogram, the weight is re-assembled for the value of the respective jet p_T as the scale:

```
COMPOSITE > ptj0_ybin0 [0,10,20,50,100,200,300,500,700,1000]
  ptj1  mu0=ptj1
  HISTOGRAM_SELECTORS
    select abs_yj1 min=0 max=.5
  END_HISTOGRAM_SELECTORS
  ptj2  mu0=ptj2
  HISTOGRAM_SELECTORS
    select abs_yj2 min=0 max=.5
  END_HISTOGRAM_SELECTORS
  ptj3  mu0=ptj3
  HISTOGRAM_SELECTORS
    select abs_yj3 min=0 max=.5
  END_HISTOGRAM_SELECTORS
  ptj4  mu0=ptj4
  HISTOGRAM_SELECTORS
    select abs_yj4 min=0 max=.5
  END_HISTOGRAM_SELECTORS
END_COMPOSITE
```

5.3.6 Scales block

The SCALES block specifies all scale combinations that should be evaluated *simultaneously* during the run. It is encapsulated by the block:

```
SCALES
  ! put scales here
END_SCALES
```

and requires at least one scale specification to be present. The first entry will be considered the central scale that is also used for the VEGAS grid adaption. Up to six additional scale specifications can be supplied. A scale specification constitutes a line in the block that specifies the factorization scale μ_F (muF) and the renormalization scale μ_R (muR)

```
muF = <sclF>  muR = <sclR>
```

In case a process that contains a fragmentation function is run, the associated factorization scale (μ_A) can be set separately using a pair muF = [<sclF>,<sclA>], which otherwise will default to $\mu_A = \mu_F$ in case only a single scale is specified. The scale <scl> can be set in two ways:

- fixed scale: <scl> = DBLE in units of [GeV];
- dynamical scale: <scl> = <obs> or DBLE*<obs>, where <obs> is the unique STR identifier of an observable.

5.3.7 Channels block

In NNLOJET a *channel* constitutes a specific scattering process, uniquely determined by the external states and the crossing, further decomposed into gauge-invariant colour factors. For example, the leading-colour $\mathcal{O}(N_c^2)$ contribution to the scattering sub-process $gg \to q\bar{q}$ constitutes one *channel* of the LO dijet production process at hadron colliders.

When the workflow described in Sect. 6.1 is used to perform the calculation, the channel information will be automatically populated, such that the user does not need to specify them

manually and instead can keep this block empty. Therefore, the information provided here is intended for expert users and/or to assist in debugging the workflow calculations.

The CHANNELS block specifies all the channels to be computed during the run by giving a list of identifiers. The VEGAS integrand is defined as the sum of the channels specified in this block (optionally grouped according to the value of the multi_channel flag, see Sect. 5.3.2). The process-id number associated with a given channel can be found in the respective driver/process/XYZ/selectchannelXYZ.f source file of the associated process XYZ. In addition, the following abbreviations are defined to simplify the specification:

- ALL;
- LO, NLO, NNLO;
- V, R;
- VV, RV, RR, RRa, RRb.

The suffices a and b for the double-real (RR) contribution only exist for specific processes and correspond to a separation of the phase space into two disjoint parts with optimized sampling strategies. The general syntax for the channel block is as follows:

```
CHANNELS region = STR
  ! put list of process id's / wildcards here
END_CHANNELS
```

In addition to specifying the sub-process on individual lines, any number of (white-space separated) process id's can be given in a single line as well. The region option only matters for the double-real contribution and processes where a phase-space separation was introduced; allowed values in this case are: a for the RRa piece, b for the RRb piece, and all (default) to include both.

5.3.8 Single-line Optional Settings

In a runcard, the user is further able to specify single-line options independent of any blocks mentioned above. All single-line settings are optional.

Reweight

The re-weighting feature has a one-line syntax

```
REWEIGHT <rwg_fct>
```

where the specification of the reweighting function <rwg_fct> follows the syntax described in Sect. 5.2.1. This function is evaluated for every event and used to re-scale the weights returned to the VEGAS integrator. It can therefore be used to increase or decrease the relevance of certain kinematic regions in the VEGAS grid adaptation, allowing for a more uniform phase-space coverage for distributions that span over several orders of magnitude. The re-weighting does not apply to the event booking into histograms. Similarly, once the VEGAS grid is frozen, i.e. during the production stage of the calculation, this option has no impact on the produced output.

PDF Interpolation

The PDF interpolation option is a single-line setting to switch on a one-dimensional cubic spline interpolation (fixed x, interpolation in μ_F), which aims to reduce the number of calls to LHAPDF. By default it is not activated but it can be switched on with the following syntax:

```
PDF_INTERPOLATION stepfac = DBLE
```

where the optional argument `stepfac` specifies the uniform spacing of the interpolation grid in the $\log(\mu_F)$ space as a multiplicative factor. The default value is 2.

6 Using NNLOJET

Following the installation steps outlined in Sect. 3.2, both the core executable (`NNLOJET`) as well as the Python workflow (`nnlojet-run`) are installed in the `bin` directory of the installation path. The main user-facing interface to NNLOJET is the workflow script `nnlojet-run`, which provides an automated way to set up and run calculations and is described in Sect. 6.1. The direct execution of the core executable is possible but requires a more in-depth knowledge of the inner workings of the program, and is therefore only recommended for expert users. Section 6.2 is therefore limited to a description of the output files that is produced by the core `NNLOJET` executable but will refrain from a detailed guide of a workflow based on it. The understanding of the output files is important in order to inspect individual runs submitted by the Python workflow in case issues are encountered. The main results are provided in the form of histogram files that are described in Sect. 6.3.

6.1 Job submission workflow

Every NNLOJET calculation (or *run*) is associated with a directory that contains all the necessary input and configuration files as well as the intermediate output and the final result. The `nnlojet-run` workflow is designed to manage the creation of the run directory, the execution of the NNLOJET core executable, and the retrieval of the final results in a user-friendly way. To this end, the workflow provides four separate sub-commands: `init`, `config`, `submit`, and `finalize`, which are described in the following. A short description of the commands and available options can be displayed by adding the `--help` command line argument:

```
$ nnlojet-run --help
$ nnlojet-run <sub-command> --help
```

6.1.1 `init`

The `init` sub-command initializes a new run directory with the necessary input and configuration files. It takes an NNLOJET runcard as input:

```
$ nnlojet-run [--exe <exe-path>] init [-o <run-path>] path/to/
   runcard
```

The command will create a new run directory at the current path named after the `<run-name>` specified in the input runcard. Optionally, the option `-o <run-path>` can be used to specify a location where the run directory should be initialized. The input runcard will be used to generate a template runcard for the calculation, which is saved as the file `template.run` in the run directory. The `NNLOJET` executable is automatically searched for in the system path but can also be set manually with the optional `--exe <exe-path>` argument.

The `init` command will first display the relevant references for the selected process that should be cited in a scientific publication using results obtained from the NNLOJET calculation; these references will also be saved as `.tex` and `.bib` files in the run directory. Subsequently, the user will be prompted to set several configuration options that are necessary for the execution:

- `policy = local|htcondor|slurm`: specifies the execution policy for how the NNLOJET calculations should be submitted. Currently supported targets are local execution

on the current machine (local) or submission on a cluster using the job scheduling systems slurm or htcondor. The cluster submission requires additional settings that are documented below. (default: local)

- order = lo|nlo|nlo_only|nnlo|nnlo_only: specifies the perturbative order of the calculation to be performed. It is also possible to restrict the calculation to only a specific coefficient, nnlo=lo+nlo_only+nnlo_only. (default: nnlo)
- target-rel-acc = DBLE: specifies the desired relative accuracy on the total cross section evaluated with the chosen fiducial cuts and for the specified order. While the specification of a REWEIGHT function can be introduced to bias the phase-space sampling, the target accuracy is always defined with respect to the unweighted cross section. (default: 0.01)
- job-max-runtime = STR|DBLE: specifies the maximum runtime for a single NNLOJET execution or a job on the cluster. The format is either a time interval specified with units (s, m, h, d, w), e.g. 1h 30m, or a floating point number in seconds. (default: 1h)
- job-fill-max-runtime = yes|no: specifies if the workflow should attempt to exhaust the maximum runtime that was set for each NNLOJET execution. This option is intended to fully utilize the wallclock time limit of cluster queues. (default: no)
- jobs-max-total = INT: specify the maximum number of NNLOJET executions that can be run in total. The workflow will terminate either once the target accuracy is reached or this limit on the total number of jobs is exhausted. (default: 100)
- jobs-max-concurrent = INT: specifies the maximum number of NNLOJET executions that can be run simultaneously. If cluster submission is selected, this option specifies the maximum number of jobs that are simultaneously submitted to the cluster. (default: 10)

If a cluster submission is selected, the user will be prompted to provide additional settings required for an execution on the respective cluster type:

- poll-time = STR|DBLE: specifies the interval in which the scheduler is queried for an update on the status of the submitted jobs. The format is either a time interval specified with units (s, m, h, d, w), e.g. 1h 30m, or a floating point number in seconds. (default: 10% of the job-max-runtime)
- submission-template = INT (index from a list displayed to the user): the job submission on a cluster relies on a submission script for the job scheduler. The workflow provides a set of built-in submission templates that can be selected by the user. The chosen template will be copied as a file <cluster>.template to the run directory and can be modified to any specific needs. Note that it is up to the user to ensure that the time limits set by the submission script, e.g. as imposed by a wallclock time limit of special queues of the cluster, are consistent with the job-max-runtime setting.

All configurations are stored in the config.json file inside the run directory, however, it is strongly advised against modifying this file manually. Instead, all interactions with the run directory should proceed through the nnlojet-run interface. The configuration of the run can be changed at any time using the config sub-command described below.

6.1.2 submit

With a properly initialized run directory at <run-path>, the main calculation can be triggered using the submit sub-command:

```
$ nnlojet-run submit <run-path>
```

This calculation will employ the options that were set during the `init` step (or alternatively overwritten using the `config` sub-command). Some of the options can also be overridden temporarily for a single submission by passing the desired settings as parameters, e.g.

```
$ nnlojet-run submit <run-path> --job-max-runtime 1h30m \
  --jobs-max-total 10 --target-rel-acc 1e-3
```

A complete list of option overrides with a short description is provided via

```
$ nnlojet-run submit --help
```

Once the `submit` sub-command is executed, the workflow is started and the process remains active for the entirety of the calculation, spawning sub-processes to execute the core `NNLOJET` program. The full calculation is divided into separate components represented by the individual lines of Eqs. (6) and (7) and further decomposed into independent partonic luminosities. The workflow proceeds by first adapting the phase-space grids in a warmup phase, followed by the actual production phase as described in Sect. 5.3.2. The warmup phase proceeds in steps, iteratively increasing the invested statistics to gradually refine the adaption until either a satisfactory grid is obtained or the specified maximum runtime is reached. The production phase optimizes the allocation of computing resources into the different parts of the calculation to minimize the final uncertainty. To this end, it is necessary to obtain an estimate on the error and the runtime per event for each piece, which is initially determined from a single low-statistics *pre-production* run. As both the warmup and the pre-production stages are necessary to proceed to the main calculation, the associated jobs do not count towards the maximum number of jobs that was set for the submission and this part of the computation will not attempt to exhaust the available resources. During the main production stage, on the other hand, the workflow will continuously dispatch new jobs in an attempt to exhaust the specified concurrent resource limits and accumulate results until either the target accuracy is reached or the maximum number of jobs is exhausted. Upon termination of all jobs, a summary of the calculation is given by printing the final cross section number with its associated uncertainty. In case the target accuracy was not reached with the allocated resources, an estimate of the number of additional jobs required to reach the requested accuracy is provided. The final results of the calculation are saved to `<run-path>/results/final`. During the workflow execution, a summary table together with logging information will be displayed to the user, see Fig. 1, which is updated in real-time as the jobs are submitted and completed.

As the `submit` command continuously monitors running jobs and dispatches new ones, it is preferrable to keep it running for the entire duration of the calculation. If necessary, it can be stopped by pressing `Ctrl+C` and re-launched at a later time. In this case, the calculation will resume from the last saved state. Jobs that are already submitted to the cluster will continue to run. If the previous execution terminated exceptionally, e.g. due to a lost `ssh` connection, the workflow will automatically attempt to recover the state and prompt the user how to proceed with the resurrection of previously active jobs.

Inbetween executions of `submit`, the settings of a run can be modified, see the `config` sub-command below. This is especially useful to increase the target accuracy, or to increase the maximal number of jobs, in case too few where initially specified. It is also possible to change the perturbative order in this way, e.g. to compute the NNLO corrections to a run that was previously calculated to NLO.

6.1.3 `finalize`

All completed jobs inside a run directory `<run-path>` can be collected and merged into a final result using the `finalize` sub-command:

#	LO	R	V	RR	RV	VV
1	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[1]	WRM A[0] D[4]	**PRD** A[1] D[0]	PRD A[0] D[1]
2	PRD A[0] D[1]	**PRD** A[1] D[0]	PRD A[0] D[1]	PRD A[0] D[1]	WRM A[1] D[3]	PRD A[0] D[1]
3	-	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[0]	WRM A[1] D[3]	PRD A[0] D[1]
4	-	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[1]
5	-	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[0]	PRD A[0] D[1]	PRD A[0] D[1]
6	-	PRD A[0] D[0]	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[1]	PRD A[0] D[1]
7	-	-	-	PRD A[0] D[0]	PRD A[0] D[1]	PRD A[0] D[1]
8	-	-	-	**PRD** A[1] D[0]	PRD A[0] D[1]	PRD A[0] D[1]
9	-	-	-	PRD A[0] D[0]	**PRD** A[1] D[0]	PRD A[0] D[1]
10	-	-	-	**PRD** A[1] D[0]	PRD A[0] D[1]	PRD A[0] D[1]
11	-	-	-	PRD A[0] D[0]	PRD A[0] D[1]	PRD A[0] D[1]
12	-	-	-	PRD A[0] D[1]	-	-
13	-	-	-	PRD A[0] D[0]	PRD A[0] D[1]	PRD A[0] D[1]
14	-	-	-	PRD A[0] D[0]	-	-
15	-	-	-	PRD A[0] D[1]	WRM A[1] D[3]	PRD A[0] D[1]
16	-	-	-	**PRD** A[1] D[0]	WRM A[1] D[3]	PRD A[0] D[1]
17	-	-	-	PRD A[0] D[0]	WRM A[1] D[3]	PRD A[0] D[1]
18	-	-	-	**PRD** A[1] D[0]	-	-
19	-	-	-	PRD A[0] D[1]	-	-
20	-	-	-	PRD A[0] D[1]	-	-
21	-	-	-	WRM A[1] D[3]	-	PRD A[0] D[1]
22	-	-	-	PRD A[0] D[1]	-	PRD A[0] D[1]
23	-	-	-	**PRD** A[1] D[0]	-	PRD A[0] D[1]
24	-	-	-	WRM A[1] D[3]	-	PRD A[0] D[1]
25	-	-	-	WRM A[1] D[3]	-	-
26	-	-	-	**PRD** A[1] D[0]	-	-
27	-	-	-	PRD A[0] D[1]	-	-

Figure 1: Example of the graphical interface to monitor the status of the calculation. Each column (LO, R, V, RR, RV, VV) represents an active component of the calculation. The indexed rows represent independent luminosity channels. In each cell of the table, the phase of calculation is indicated: warmup ("WRM") or production ("PRD"). The number n_A of active jobs is indicated as `A[`n_A`]`, which includes both pending and actively running jobs, while the number n_D of completed jobs is indicated as `D[`n_D`]`. In case of unsuccessful NNLOJET executions, the cell will additionally include an entry `F[`n_F`]`, indicating the number of failed jobs n_F.

```
$ nnlojet-run finalize <run-path>
```

It should be noted that the same combination procedure is also triggered at the end of the `submit` sub-command and it is therefore not necessary to explicitly execute a `finalize` after a `submit`.

The `finalize` sub-command facilitates a manual trigger to perform a re-combination of results with changed internal parameters of the merging algorithm. The main purpose of the merging algorithm is to combine the partial results of different parts of the calculation into a final prediction and to mitigate the impact from outliers. Potential issues pertaining to the treatment of outliers, as well as the approach employed in our implementation are presented in Ref. [147]. The underlying algorithm is controlled by a set of parameters that have sensible defaults but can also be overridden by the user. This is useful to check if the combine step is potentially introducing a systematic bias as well as to optimize the combination to obtain the best possible result from the raw data. The merging proceeds in steps with a dynamical termination condition, which are applied for each histogram separately on a *bin-by-bin* basis:

1. trim: apply an outlier-rejection procedure based on the inter-quartile-range to discard data points which are identified as severe outliers;

2. k-scan: on the trimmed dataset, merge pairs of (pseudo-)jobs in an unweighted manner into a new pseudo-job, which is statistically equivalent to running a single job with the

combined statistics;

3. weighted combination: all pseudo-jobs are merged using a weighted average to further suppress outliers and the result is stored;
4. repeat steps 2 & 3 until `nsteps` results from weighted combinations are accumulated;
5. termination: check for a plateau in the last `nsteps` results; if no plateau is found, go back to step 2.

The core strategy is to combine as many individual jobs into pseudo-jobs as necessary to ensure that the resulting statistical uncertainty for the pseudo-job is reliable and therefore the weighted average can be safely taken in the next step to further suppress the impact of outliers.

The parameters controlling the trimming step are:

- `trim-threshold = DBLE`: a threshold value for the distance from the median in units of the inter-quartile-distance beyond which a data point will be considered an outlier and is discarded. Larger values for this parameter reduce the number of data points that are discarded. (default = `3.5`)
- `trim-max-fraction = DBLE`: a safeguard on the maximal fraction of data points that are allowed to be trimmed away. For each bin in violation of this condition, its threshold value is dynamically increased until the ratio of trimmed data in that bin falls below this value. (default = `0.01`)

The termination condition of identifying a plateau (the k-scan) is controlled by the following parameters:

- `k-scan-nsteps = INT`: the number of pair-combination steps across which the pleateau is checked. (default: `2`)
- `k-scan-maxdev-steps = DBLE`: the tolerance for the plateau condition in units of the standard deviation. Across the last `k-scan-nsteps` steps of the k-scan, require that all points are mutually compatible within this tolerance. Note that the default value is smaller than 1 as the individual points from the k-scan are determined from the *same* dataset and thus are strongly correlated. (default = `0.5`)

The default values for this parameter can be set using the `config` sub-command described below. Alternatively, these options can also be overridden temporarily for a single `finalize` step by passing the desired settings as options, e.g.

```
$ nnlojet-run finalize <run-path> --trim-threshold 5 \
  --k-scan-nsteps 3 --k-scan-maxdev-steps 0.1
```

which is documented in the corresponding help output: `nnlojet-run finalize --help`.

6.1.4 `config`

All default configuration settings can be changed at a later time using the `config` sub-command:

```
$ nnlojet-run config <run-path>
```

This will open an interactive prompt that allows the user to change the configuration settings that were set during the `init` step. The merge settings used for the `finalize` sub-command can instead be overwritten using:

```
$ nnlojet-run config <run-path> --merge
```

In addition, advanced settings of the workflow can be viewed and set with

```
$ nnlojet-run config <run-path> --advanced
```

which provides, e.g. the possibility to change the seed offset of the workflow.

6.1.5 Run directory

The run directory is structured as follows:

- `NNLOJET_references_<PROC>.[bib|tex]` for the references to be cited in publications;
- `config.json`: contains all the configuration settings for the run;
- `db.sqlite`: a database file that contains the status of the run (process and job information);
- `log.sqlite`: a database file that contains the logging information of the submission;
- `template.run`: the template runcard for the calculation;
- `raw` directory: contains the raw output of the NNLOJET runs saved in two subdirectories (`warmup`, `production`) with the individual job results;
- `results` directory: contains the final results of the calculation;
  - `final` directory: contains the final results in the form of data files `<order>.<obs>.dat` for all complete orders;
  - `merge` directory: contains the result of the currently active order;
  - `parts` directory: all intermediate combined results separately for each part of the calculation;

The file format of the histogram data files is described below in Sect. 6.3.

6.2 Direct execution

The direct execution of the core `NNLOJET` executable produces output files with the following naming conventions:

- Log files: `<proc>.<run>.s<seed>.log`;
- Grid files (machine readable): `<proc>.<run>.y<t-cut>.<cont>`;
- Grid files (human readable): `<proc>.<run>.y<t-cut>.<cont>_iterations.txt`;
- Histogram files: `<proc>.<run>.<cont>.<obs>.s<seed>.dat`;

where `<proc>` denotes the process name, `<run>` the run name, `<seed>` the seed number, `<cont>` the contribution (LO, R, V, ...), and `<obs>` the observable name.

6.3 Histogram data files

The data files produced by NNLOJET are in a simple text-based format with units of [fb] for cross sections and [GeV] for dimensionful quantities. The first n_x columns of the files are reserved for the binning information, if necessary:

- cross section data files have no bins and thus $n_x = 0$;
- differential distributions have $n_x = 3$ for the lower edge, centre, and upper edge of each bin;

- cumulant distributions have $n_x = 1$ for the integration limit for the cumulant.

The remaining columns are composed of a series of pairs of numbers that represent the value and the associated Monte Carlo integration uncertainty. First, these pairs are written out for all the active scales, as specified in the `SCALES` block and its order. Depending on the `output_type` specified for the histograms, this is optionally repeated with the same pattern for the separate partonic luminosities.

A description of the individual columns is also provided as a comment line starting with `#labels`. In addition, the cross section contribution from events that fall outside of chosen histogram range are collected and written out in the `#overflow` line following the same column structure.

7 Conclusion

This paper accompanies the open-source release of the parton-level event generator NNLOJET. The NNLOJET code is distributed under the GPLv3.0 license and can be downloaded from HEPForge: http://nnlojet.hepforge.org.

NNLOJET is steered through a runcard whose syntax is documented in this paper. Some example runcards are distributed with the release and more examples with sample output data are provided via http://nnlojet.hepforge.org/examples.html. The NNLOJET program can either be executed directly or through a workflow script that is part of the distribution and which dynamically adapts the usage of computing resources towards a pre-specified target accuracy. We described the usage of the `NNLOJET` executable and the `nnlojet-run` workflow script in detail.

In its current release (v1.0.0), NNLOJET contains a range of lepton- and hadron-collider jet production processes of $2 \to 2$ and $1 \to 3$ kinematics. More processes and features will be added in future releases and the code documentation on http://nnlojet.hepforge.org will be kept up to date to reflect those changes. User feedback and bug reports should be addressed to: nnlojet-support@cern.ch.

Acknowledgements

The NNLOJET project has been enabled by substantial support throughout the past decade from the European Research Council (ERC) through grant agreements 340983 (ERC Advanced Grant MC@NNLO) and 101019620 (ERC Advanced Grant TOPUP), from the Swiss National Science Foundation (SNF) under contracts 200020-162487, 200021-172478, 200020-175595, 200021-197130, 200020-204200 and 200021-231259, from the Swiss National Supercomputing Centre (CSCS) under project IDs ETH5f and UZH10, from the Royal Society, from the UK Science and Technology Facilities Council (STFC) under contracts ST/G000905/1, ST/P001246/1, ST/T001011/1 and ST/X000745/1, and from the National Science Foundation of China (NSFC) under contract No.12475085, No.12321005 and No.1223500. MM is supported by a Royal Society Newton International Fellowship (NIF/R1/232539). JP acknowledges Fundação para a Ciência e a Tecnologia (FCT) for its financial support through the programatic funding of R&D units (contract UIDP/50007/2020) and under project CERN/FIS-PAR/0032/2021. The work of LB, XC, GF, MJ, and KS was supported by the UZH Forschungskredit Candoc and Postdoc schemes.

A Example: $Z + 1j$ at NNLO

In this Appendix, we highlight an example calculation for the $Z+1j$ process at NNLO. The full runcard is provided below, implementing the following items:

- In the PROCESS block, we specify the process (Z-boson plus one jet) and the collider (proton–proton collider at $\sqrt{s} = 7\,\text{TeV}$. We further specify the jet algorithm used to reconstruct jets (anti-k_t with $R = 0.4$). See Sect. 5.3.1.
- In the RUN block, we specify the PDF set, here NNPDF3.1. Additionally, we indicate a value for the technical cutoff (equal to the default $t_{\text{cut}} = 10^{-8}$) and we turn on the multi-channel adapatation over individual channels (in this case, channels will be grouped such that every phase-space point will evaluate a subset of 5 channels). See Sect. 5.3.2.
- In the PARAMETERS block, we select the G_μ scheme and set the values for the masses and widths of the Z and W bosons and for the Fermi constant G_F. See Sect. 5.3.3.
- In the SELECTORS block, we define the fiducial phase space for the selection of events. In particular, we require leptons to have $p_{\text{T}} > 20\,\text{GeV}$ and $|y| < 2.5$. Moreover, their angular distance in the rapidity–azimuth plane should be $\Delta R_{\ell\ell} > 0.2$ and they should feature an invariant mass $m_{\ell\ell} \in [66, 116]\,\text{GeV}$. Jets are required to have $p_{\text{T}} > 30\,\text{GeV}$ and $|y| < 4.4$, with $\Delta R_{\ell j} > 0.5$. See Sect. 5.3.4.
- In the HISTOGRAMS block, we declare some histograms, such as the transverse momentum or the absolute rapidity of the leading-p_{T} jet. Note in particular some features like: the possibility of specifying uniformly-spaced bins or declaring them with a custom list; the file-name override (`ptj1 > ptj1_exc`) to avoid name clashes when saving histograms; the histogram selectors to generate multi-differential distributions (e.g. we fill the `ptz_y1` histogram with the transverse momentum of the Z-boson only when $0.5 < |y_Z| < 1.0$). See Sect. 5.3.5.
- In the SCALES block, we specify the set of renormalization and factorization scales to be registered for the calculation: we adopt $\mu_R = \mu_F = \frac{1}{2}H_T$ (with H_T the scalar-p_{T} sum over leptons and jets) as central scale and we further compute six additional combinations of μ_R and μ_F according to the usual 7-point scale variation prescription to estimate theory uncertanties. See Sect. 5.3.6.
- The CHANNELS block is left empty as it will be automatically populated by the workflow described in Sect. 6.1. See Sect. 5.3.7.

```
PROCESS ZJ
  collider = pp
  sqrts = 7000.0
  jet = antikt[0.4]
END_PROCESS

RUN EXZJ
  PDF = NNPDF31_nnlo_as_0118[0]
  tcut = 1d-8
  multi_channel = 5
END_RUN

PARAMETERS
  MASS[Z] = 91.1876
  WIDTH[Z] = 2.4952
```

```
  MASS[W] = 80.385
  WIDTH[W] = 2.085
  SCHEME[alpha] = Gmu
  GF = 1.1663787d-5
END_PARAMETERS

SELECTORS
  select ptlm            min = 20
  select ptlp            min = 20
  select abs_ylm         max = 2.5
  select abs_ylp         max = 2.5
  select dr_l12          min = 0.2
  select mll             min = 66   max = 116
  select jets_pt         min = 30
  select jets_abs_y      max = 4.4
  select min_dr_lj       min = 0.5
END_SELECTORS

HISTOGRAMS
  njets [-0.5,0.5,1.5,2.5,3.5]
  ptj1 [30,40,60,80,120,160,200,250,310,370,450,550,700]
  abs_yj1 [0,0.4,0.8,1.2,1.6,2.0,2.4,2.8,3.4,4.4]
  ptj1 > ptj1_exc [30,40,60,80,120,160,200,250,310,370,450]
  HISTOGRAM_SELECTORS
    select njets min = 1 max = 1
  END_HISTOGRAM_SELECTORS
  ptz min = 30 max = 330 nbins = 10
  ptz > ptz_y1 min = 30 max = 330 nbins = 10
  HISTOGRAM_SELECTORS
    select abs_yz min = 0.5  max = 1.0
  END_HISTOGRAM_SELECTORS
END_HISTOGRAMS

SCALES
  muf = 0.50 * ht_full    mur = 0.50 * ht_full
  muf = 0.25 * ht_full    mur = 0.50 * ht_full
  muf = 0.50 * ht_full    mur = 0.25 * ht_full
  muf = 0.25 * ht_full    mur = 0.25 * ht_full
  muf = 1.00 * ht_full    mur = 0.50 * ht_full
  muf = 0.50 * ht_full    mur = 1.00 * ht_full
  muf = 1.00 * ht_full    mur = 1.00 * ht_full
END_SCALES

CHANNELS
END_CHANNELS
```

The above runcard can be used as input to the job-submission workflow described in Sect. 6.1. After a run on the CERN lxplus submission system with a total runtime of order 50k CPU hours, one obtains a value of 68.5(2) pb for the total NNLO fiducial cross section. Figure 2 shows an example of a differential distribution, namely the transverse momentum of the leading jet. The runcard with all generated data files and plots are included as ancillary files to this publication.

We remind the reader that more example runcards, including reference results, are available on the website https://nnlojet.hepforge.org/examples.html.

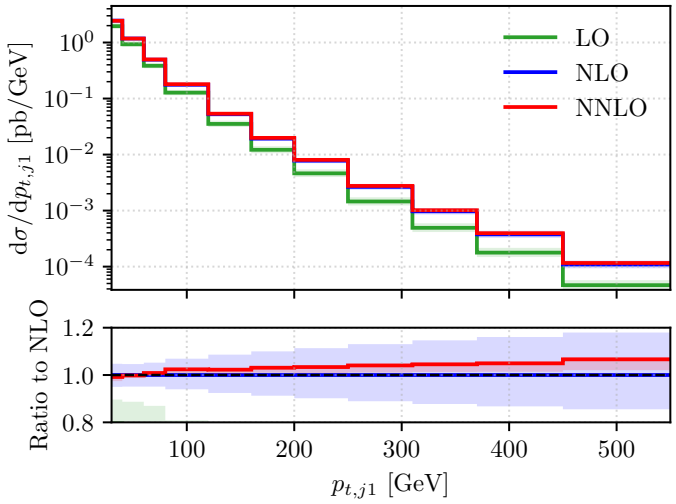


Figure 2: Differential distributions at LO, NLO, and NNLO for the transverse momentum of the leading jet in $Z + 1j$. The uncertainty bands represent the envelope of a 7-point scale variation.

References

[1] R. K. Ellis, W. J. Stirling and B. R. Webber, *QCD and collider physics*, vol. 8, Cambridge University Press, ISBN 978-0-511-82328-2, 978-0-521-54589-1, doi:10.1017/CBO9780511628788 (2011).

[2] J. Campbell, J. Huston and F. Krauss, *The Black Book of Quantum Chromodynamics: a Primer for the LHC Era*, Oxford University Press, ISBN 978-0-19-965274-7, doi:10.1093/oso/9780199652747.001.0001 (2018).

[3] G. F. Sterman and S. Weinberg, *Jets from Quantum Chromodynamics*, Phys. Rev. Lett. **39**, 1436 (1977), doi:10.1103/PhysRevLett.39.1436.

[4] S. Catani, *The Singular behavior of QCD amplitudes at two loop order*, Phys. Lett. B **427**, 161 (1998), doi:10.1016/S0370-2693(98)00332-3, [hep-ph/9802439].

[5] T. Becher and M. Neubert, *Infrared singularities of scattering amplitudes in perturbative QCD*, Phys. Rev. Lett. **102**, 162001 (2009), doi:10.1103/PhysRevLett.102.162001, [Erratum: Phys.Rev.Lett. 111, 199905 (2013)], [0901.0722].

[6] E. Gardi and L. Magnea, *Factorization constraints for soft anomalous dimensions in QCD scattering amplitudes*, JHEP **03**, 079 (2009), doi:10.1088/1126-6708/2009/03/079, [0901.1091].

[7] L. J. Dixon, *Matter Dependence of the Three-Loop Soft Anomalous Dimension Matrix*, Phys. Rev. D **79**, 091501 (2009), doi:10.1103/PhysRevD.79.091501, [0901.3414].

[8] W. T. Giele, E. W. N. Glover and D. A. Kosower, *Higher order corrections to jet cross-sections in hadron colliders*, Nucl. Phys. B **403**, 633 (1993), doi:10.1016/0550-3213(93)90365-V, [hep-ph/9302225].

[9] S. Catani and M. H. Seymour, *A General algorithm for calculating jet cross-sections in NLO QCD*, Nucl. Phys. B **485**, 291 (1997), doi:10.1016/S0550-3213(96)00589-5, [Erratum: Nucl.Phys.B 510, 503–504 (1998)], [hep-ph/9605323].

[10] S. Frixione, Z. Kunszt and A. Signer, *Three jet cross-sections to next-to-leading order*, Nucl. Phys. B **467**, 399 (1996), doi:10.1016/0550-3213(96)00110-1, [hep-ph/9512328].

[11] G. Ossola, C. G. Papadopoulos and R. Pittau, *CutTools: A Program implementing the OPP reduction method to compute one-loop amplitudes*, JHEP **03**, 042 (2008), doi:10.1088/1126-6708/2008/03/042, [0711.3596].

[12] R. K. Ellis, Z. Kunszt, K. Melnikov and G. Zanderighi, *One-loop calculations in quantum field theory: from Feynman diagrams to unitarity cuts*, Phys. Rept. **518**, 141 (2012), doi:10.1016/j.physrep.2012.01.008, [1105.4319].

[13] F. Cascioli, P. Maierhöfer and S. Pozzorini, *Scattering Amplitudes with Open Loops*, Phys. Rev. Lett. **108**, 111601 (2012), doi:10.1103/PhysRevLett.108.111601, [1111.5206].

[14] G. Cullen *et al.*, *GoSam-2.0: a tool for automated one-loop calculations within the Standard Model and beyond*, Eur. Phys. J. C **74**(8), 3001 (2014), doi:10.1140/epjc/s10052-014-3001-5, [1404.7096].

[15] S. Alioli, P. Nason, C. Oleari and E. Re, *A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX*, JHEP **06**, 043 (2010), doi:10.1007/JHEP06(2010)043, [1002.2581].

[16] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H. S. Shao, T. Stelzer, P. Torrielli and M. Zaro, *The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations*, JHEP **07**, 079 (2014), doi:10.1007/JHEP07(2014)079, [1405.0301].

[17] J. Bellm *et al.*, *Herwig 7.0/Herwig++ 3.0 release note*, Eur. Phys. J. C **76**(4), 196 (2016), doi:10.1140/epjc/s10052-016-4018-8, [1512.01178].

[18] E. Bothmann *et al.*, *Event Generation with Sherpa 2.2*, SciPost Phys. **7**(3), 034 (2019), doi:10.21468/SciPostPhys.7.3.034, [1905.09127].

[19] C. Bierlich *et al.*, *A comprehensive guide to the physics and usage of PYTHIA 8.3*, SciPost Phys. Codeb. **2022**, 8 (2022), doi:10.21468/SciPostPhysCodeb.8, [2203.11601].

[20] J. Campbell and T. Neumann, *Precision Phenomenology with MCFM*, JHEP **12**, 034 (2019), doi:10.1007/JHEP12(2019)034, [1909.09117].

[21] K. Arnold *et al.*, *VBFNLO: A Parton level Monte Carlo for processes with electroweak bosons*, Comput. Phys. Commun. **180**, 1661 (2009), doi:10.1016/j.cpc.2009.03.006, [0811.4559].

[22] J. M. Campbell and E. W. N. Glover, *Double unresolved approximations to multiparton scattering amplitudes*, Nucl. Phys. B **527**, 264 (1998), doi:10.1016/S0550-3213(98)00295-8, [hep-ph/9710255].

[23] S. Catani and M. Grazzini, *Infrared factorization of tree level QCD amplitudes at the next-to-next-to-leading order and beyond*, Nucl. Phys. B **570**, 287 (2000), doi:10.1016/S0550-3213(99)00778-6, [hep-ph/9908523].

[24] S. Catani and M. Grazzini, *The soft gluon current at one loop order*, Nucl. Phys. B **591**, 435 (2000), doi:10.1016/S0550-3213(00)00572-1, [hep-ph/0007142].

[25] D. A. Kosower and P. Uwer, *One loop splitting amplitudes in gauge theory*, Nucl. Phys. B **563**, 477 (1999), doi:10.1016/S0550-3213(99)00583-0, [hep-ph/9903515].

[26] Z. Bern, V. Del Duca, W. B. Kilgore and C. R. Schmidt, *The infrared behavior of one loop QCD amplitudes at next-to-next-to leading order*, Phys. Rev. D **60**, 116001 (1999), doi:10.1103/PhysRevD.60.116001, [hep-ph/9903516].

[27] T. Binoth and G. Heinrich, *Numerical evaluation of phase space integrals by sector decomposition*, Nucl. Phys. B **693**, 134 (2004), doi:10.1016/j.nuclphysb.2004.06.005, [hep-ph/0402265].

[28] C. Anastasiou, K. Melnikov and F. Petriello, *A new method for real radiation at NNLO*, Phys. Rev. D **69**, 076010 (2004), doi:10.1103/PhysRevD.69.076010, [hep-ph/0311311].

[29] S. Catani and M. Grazzini, *An NNLO subtraction formalism in hadron collisions and its application to Higgs boson production at the LHC*, Phys. Rev. Lett. **98**, 222002 (2007), doi:10.1103/PhysRevLett.98.222002, [hep-ph/0703012].

[30] A. Gehrmann-De Ridder, T. Gehrmann and E. W. N. Glover, *Antenna subtraction at NNLO*, JHEP **09**, 056 (2005), doi:10.1088/1126-6708/2005/09/056, [hep-ph/0505111].

[31] M. Czakon, *A novel subtraction scheme for double-real radiation at NNLO*, Phys. Lett. B **693**, 259 (2010), doi:10.1016/j.physletb.2010.08.036, [1005.0274].

[32] R. Boughezal, K. Melnikov and F. Petriello, *A subtraction scheme for NNLO computations*, Phys. Rev. D **85**, 034025 (2012), doi:10.1103/PhysRevD.85.034025, [1111.7041].

[33] R. Boughezal, X. Liu and F. Petriello, *N-jettiness soft function at next-to-next-to-leading order*, Phys. Rev. D **91**(9), 094035 (2015), doi:10.1103/PhysRevD.91.094035, [1504.02540].

[34] J. Gaunt, M. Stahlhofen, F. J. Tackmann and J. R. Walsh, *N-jettiness Subtractions for NNLO QCD Calculations*, JHEP **09**, 058 (2015), doi:10.1007/JHEP09(2015)058, [1505.04794].

[35] M. Cacciari, F. A. Dreyer, A. Karlberg, G. P. Salam and G. Zanderighi, *Fully Differential Vector-Boson-Fusion Higgs Production at Next-to-Next-to-Leading Order*, Phys. Rev. Lett. **115**(8), 082002 (2015), doi:10.1103/PhysRevLett.115.082002, [Erratum: Phys.Rev.Lett. 120, 139901 (2018)], [1506.02660].

[36] F. Caola, K. Melnikov and R. Röntsch, *Nested soft-collinear subtractions in NNLO QCD computations*, Eur. Phys. J. C **77**(4), 248 (2017), doi:10.1140/epjc/s10052-017-4774-0, [1702.01352].

[37] V. Del Duca, C. Duhr, A. Kardos, G. Somogyi, Z. Szőr, Z. Trócsányi and Z. Tulipánt, *Jet production in the CoLoRFulNNLO method: event shapes in electron-positron collisions*, Phys. Rev. D **94**(7), 074019 (2016), doi:10.1103/PhysRevD.94.074019, [1606.03453].

[38] L. Magnea, G. Pelliccioli, C. Signorile-Signorile, P. Torrielli and S. Uccirati, *Analytic integration of soft and collinear radiation in factorised QCD cross sections at NNLO*, JHEP **02**, 037 (2021), doi:10.1007/JHEP02(2021)037, [2010.14493].

[39] G. Heinrich, *Collider Physics at the Precision Frontier*, Phys. Rept. **922**, 1 (2021), doi:10.1016/j.physrep.2021.03.006, [2009.00516].

[40] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover and G. Heinrich, *EERAD3: Event shapes and jet rates in electron-positron annihilation at order α_s^3*, Comput. Phys. Commun. **185**, 3331 (2014), doi:10.1016/j.cpc.2014.07.024, [1402.4140].

[41] B. C. Aveleira, A. Gehrmann-De Ridder, T. Gehrmann, N. Glover, G. Heinrich and C. T. Preuss, *EERAD3 version 2: QCD corrections in hadronic colour-singlet decays*, SciPost Phys. Codeb. **59**, 1 (2025), doi:10.21468/SciPostPhysCodeb.59, [2503.20610].

[42] M. Grazzini, S. Kallweit and M. Wiesemann, *Fully differential NNLO computations with MATRIX*, Eur. Phys. J. C **78**(7), 537 (2018), doi:10.1140/epjc/s10052-018-5771-7, [1711.06631].

[43] P. F. Monni, P. Nason, E. Re, M. Wiesemann and G. Zanderighi, *MiNNLO$_{PS}$: a new method to match NNLO QCD to parton showers*, JHEP **05**, 143 (2020), doi:10.1007/JHEP05(2020)143, [Erratum: JHEP 02, 031 (2022)], [1908.06987].

[44] P. F. Monni, E. Re and M. Wiesemann, *MiNNLO$_{PS}$: optimizing $2 \to 1$ hadronic processes*, Eur. Phys. J. C **80**(11), 1075 (2020), doi:10.1140/epjc/s10052-020-08658-5, [2006.04133].

[45] J. M. Campbell, R. K. Ellis and S. Seth, *Non-local slicing approaches for NNLO QCD in MCFM*, JHEP **06**, 002 (2022), doi:10.1007/JHEP06(2022)002, [2202.07738].

[46] C. Anastasiou, L. J. Dixon, K. Melnikov and F. Petriello, *Dilepton rapidity distribution in the Drell-Yan process at NNLO in QCD*, Phys. Rev. Lett. **91**, 182002 (2003), doi:10.1103/PhysRevLett.91.182002, [hep-ph/0306192].

[47] S. Catani, L. Cieri, G. Ferrera, D. de Florian and M. Grazzini, *Vector boson production at hadron colliders: a fully exclusive QCD calculation at NNLO*, Phys. Rev. Lett. **103**, 082001 (2009), doi:10.1103/PhysRevLett.103.082001, [0903.2120].

[48] R. Gavin, Y. Li, F. Petriello and S. Quackenbush, *FEWZ 2.0: A code for hadronic Z production at next-to-next-to-leading order*, Comput. Phys. Commun. **182**, 2388 (2011), doi:10.1016/j.cpc.2011.06.008, [1011.3540].

[49] S. Catani, L. Cieri, D. de Florian, G. Ferrera and M. Grazzini, *Diphoton production at hadron colliders: a fully-differential QCD calculation at NNLO*, Phys. Rev. Lett. **108**, 072001 (2012), doi:10.1103/PhysRevLett.108.072001, [Erratum: Phys.Rev.Lett. 117, 089901 (2016)], [1110.2375].

[50] S. Camarda *et al.*, *DYTurbo: Fast predictions for Drell-Yan processes*, Eur. Phys. J. C **80**(3), 251 (2020), doi:10.1140/epjc/s10052-020-7757-5, [Erratum: Eur.Phys.J.C 80, 440 (2020)], [1910.07049].

[51] V. Del Duca, C. Duhr, L. Fekeshazy, F. Guadagni, P. Mukherjee, G. Somogyi, F. Tramontano and S. Van Thurenhout, *NNLOCAL: completely local subtractions for color-singlet production in hadron collisions*, JHEP **05**, 151 (2025), doi:10.1007/JHEP05(2025)151, [2412.21028].

[52] E. Bothmann *et al.*, *Event generation with Sherpa 3*, JHEP **12**, 156 (2024), doi:10.1007/JHEP12(2024)156, [2410.22148].

[53] A. Daleo, T. Gehrmann and D. Maitre, *Antenna subtraction with hadronic initial states*, JHEP **04**, 016 (2007), doi:10.1088/1126-6708/2007/04/016, [hep-ph/0612257].

[54] J. Currie, E. W. N. Glover and S. Wells, *Infrared Structure at NNLO Using Antenna Subtraction*, JHEP **04**, 066 (2013), doi:10.1007/JHEP04(2013)066, [1301.4693].

[55] A. Gehrmann-De Ridder, T. Gehrmann and E. W. N. Glover, *Infrared structure of $e^+e^- \to 2$ jets at NNLO*, Nucl. Phys. B **691**, 195 (2004), doi:10.1016/j.nuclphysb.2004.05.017, [hep-ph/0403057].

[56] A. Gehrmann-De Ridder, T. Gehrmann and E. W. N. Glover, *Quark-gluon antenna functions from neutralino decay*, Phys. Lett. B **612**, 36 (2005), doi:10.1016/j.physletb.2005.02.039, [hep-ph/0501291].

[57] A. Gehrmann-De Ridder, T. Gehrmann and E. W. N. Glover, *Gluon-gluon antenna functions from Higgs boson decay*, Phys. Lett. B **612**, 49 (2005), doi:10.1016/j.physletb.2005.03.003, [hep-ph/0502110].

[58] D. A. Kosower, *Multiple singular emission in gauge theories*, Phys. Rev. D **67**, 116003 (2003), doi:10.1103/PhysRevD.67.116003, [hep-ph/0212097].

[59] A. Daleo, A. Gehrmann-De Ridder, T. Gehrmann and G. Luisoni, *Antenna subtraction at NNLO with hadronic initial states: initial-final configurations*, JHEP **01**, 118 (2010), doi:10.1007/JHEP01(2010)118, [0912.0374].

[60] T. Gehrmann and P. F. Monni, *Antenna subtraction at NNLO with hadronic initial states: real-virtual initial-initial configurations*, JHEP **12**, 049 (2011), doi:10.1007/JHEP12(2011)049, [1107.4037].

[61] R. Boughezal, A. Gehrmann-De Ridder and M. Ritzmann, *Antenna subtraction at NNLO with hadronic initial states: double real radiation for initial-initial configurations with two quark flavours*, JHEP **02**, 098 (2011), doi:10.1007/JHEP02(2011)098, [1011.6631].

[62] A. Gehrmann-De Ridder, T. Gehrmann and M. Ritzmann, *Antenna subtraction at NNLO with hadronic initial states: double real initial-initial configurations*, JHEP **10**, 047 (2012), doi:10.1007/JHEP10(2012)047, [1207.5779].

[63] X. Chen, T. Gehrmann, E. W. N. Glover, A. Huss and M. Marcoli, *Automation of antenna subtraction in colour space: gluonic processes*, JHEP **10**, 099 (2022), doi:10.1007/JHEP10(2022)099, [2203.13531].

[64] T. Gehrmann, E. W. N. Glover and M. Marcoli, *The colourful antenna subtraction method*, JHEP **03**, 114 (2024), doi:10.1007/JHEP03(2024)114, [2310.19757].

[65] T. Gehrmann *et al.*, *Jet cross sections and transverse momentum distributions with NNLO-JET*, PoS **RADCOR2017**, 074 (2018), doi:10.22323/1.290.0074, [1801.06415].

[66] G. P. Lepage, *A New Algorithm for Adaptive Multidimensional Integration*, J. Comput. Phys. **27**, 192 (1978), doi:10.1016/0021-9991(78)90004-9.

[67] E. W. N. Glover and J. Pires, *Antenna subtraction for gluon scattering at NNLO*, JHEP **06**, 096 (2010), doi:10.1007/JHEP06(2010)096, [1003.2824].

[68] J. M. Campbell and R. K. Ellis, *Next-to-Leading Order Corrections to $W + 2$ Jet and $Z + 2$ Jet Production at Hadron Colliders*, Phys. Rev. D **65**, 113007 (2002), doi:10.1103/PhysRevD.65.113007, [hep-ph/0202176].

[69] E. Remiddi and J. A. M. Vermaseren, *Harmonic polylogarithms*, Int. J. Mod. Phys. A **15**, 725 (2000), doi:10.1142/S0217751X00000367, [hep-ph/9905237].

[70] T. Gehrmann and E. Remiddi, *Numerical evaluation of harmonic polylogarithms*, Comput. Phys. Commun. **141**, 296 (2001), doi:10.1016/S0010-4655(01)00411-8, [hep-ph/0107173].

[71] T. Gehrmann and E. Remiddi, *Numerical evaluation of two-dimensional harmonic polylogarithms*, Comput. Phys. Commun. **144**, 200 (2002), doi:10.1016/S0010-4655(02)00139-X, [hep-ph/0111255].

[72] S. Bühler and C. Duhr, *CHAPLIN - Complex Harmonic Polylogarithms in Fortran*, Comput. Phys. Commun. **185**, 2703 (2014), doi:10.1016/j.cpc.2014.05.022, [1106.5739].

[73] A. Buckley, J. Ferrando, S. Lloyd, K. Nordström, B. Page, M. Rüfenacht, M. Schönherr and G. Watt, *LHAPDF6: parton density access in the LHC precision era*, Eur. Phys. J. C **75**, 132 (2015), doi:10.1140/epjc/s10052-015-3318-8, [1412.7420].

[74] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover and G. Heinrich, *NNLO corrections to event shapes in e^+e^- annihilation*, JHEP **12**, 094 (2007), doi:10.1088/1126-6708/2007/12/094, [0711.4711].

[75] T. Gehrmann, E. W. N. Glover, A. Huss, J. Niehues and H. Zhang, *NNLO QCD corrections to event orientation in e^+e^- annihilation*, Phys. Lett. B **775**, 185 (2017), doi:10.1016/j.physletb.2017.10.069, [1709.01097].

[76] K. Hagiwara and D. Zeppenfeld, *Amplitudes for Multiparton Processes Involving a Current at e^+e^-, $e^\pm p$ and Hadron Colliders*, Nucl. Phys. B **313**, 560 (1989), doi:10.1016/0550-3213(89)90397-0.

[77] F. A. Berends, W. T. Giele and H. Kuijf, *Exact Expressions for Processes Involving a Vector Boson and Up to Five Partons*, Nucl. Phys. B **321**, 39 (1989), doi:10.1016/0550-3213(89)90242-3.

[78] Z. Bern, L. J. Dixon and D. A. Kosower, *One loop amplitudes for e^+e^- to four partons*, Nucl. Phys. B **513**, 3 (1998), doi:10.1016/S0550-3213(97)00703-7, [hep-ph/9708239].

[79] L. W. Garland, T. Gehrmann, E. W. N. Glover, A. Koukoutsakis and E. Remiddi, *The Two loop QCD matrix element for $e^+e^- \to 3$ jets*, Nucl. Phys. B **627**, 107 (2002), doi:10.1016/S0550-3213(02)00057-3, [hep-ph/0112081].

[80] L. W. Garland, T. Gehrmann, E. W. N. Glover, A. Koukoutsakis and E. Remiddi, *Two loop QCD helicity amplitudes for $e^+e^- \to 3$ jets*, Nucl. Phys. B **642**, 227 (2002), doi:10.1016/S0550-3213(02)00627-2, [hep-ph/0206067].

[81] T. Gehrmann and E. Remiddi, *Analytic continuation of massless two loop four point functions*, Nucl. Phys. B **640**, 379 (2002), doi:10.1016/S0550-3213(02)00569-2, [hep-ph/0207020].

[82] T. Gehrmann and L. Tancredi, *Two-loop QCD helicity amplitudes for $q\bar{q} \to W^\pm\gamma$ and $q\bar{q} \to Z^0\gamma$*, JHEP **02**, 004 (2012), doi:10.1007/JHEP02(2012)004, [1112.1531].

[83] O. Biebel, *Experimental tests of the strong interaction and its energy dependence in electron positron annihilation*, Phys. Rept. **340**, 165 (2001), doi:10.1016/S0370-1573(00)00072-7.

[84] J. Currie, T. Gehrmann, A. Huss and J. Niehues, *NNLO QCD corrections to jet production in deep inelastic scattering*, JHEP **07**, 018 (2017), doi:10.1007/JHEP07(2017)018, [Erratum: JHEP 12, 042 (2020)], [1703.05977].

[85] J. Niehues and D. M. Walker, *NNLO QCD Corrections to Jet Production in Charged Current Deep Inelastic Scattering*, Phys. Lett. B **788**, 243 (2019), doi:10.1016/j.physletb.2018.11.025, [1807.02529].

[86] T. Gehrmann, A. Huss, J. Mo and J. Niehues, *Second-order QCD corrections to event shape distributions in deep inelastic scattering*, Eur. Phys. J. C **79**(12), 1022 (2019), doi:10.1140/epjc/s10052-019-7528-3, [1909.02760].

[87] A. Gehrmann-De Ridder, E. W. N. Glover and J. Pires, *Real-Virtual corrections for gluon scattering at NNLO*, JHEP **02**, 141 (2012), doi:10.1007/JHEP02(2012)141, [1112.3613].

[88] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover and J. Pires, *Double Virtual corrections for gluon scattering at NNLO*, JHEP **02**, 026 (2013), doi:10.1007/JHEP02(2013)026, [1211.2710].

[89] J. Currie, E. W. N. Glover and J. Pires, *Next-to-Next-to Leading Order QCD Predictions for Single Jet Inclusive Production at the LHC*, Phys. Rev. Lett. **118**(7), 072002 (2017), doi:10.1103/PhysRevLett.118.072002, [1611.01460].

[90] J. Currie, A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and J. Pires, *Precise predictions for dijet production at the LHC*, Phys. Rev. Lett. **119**(15), 152001 (2017), doi:10.1103/PhysRevLett.119.152001, [1705.10271].

[91] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and J. Pires, *Triple Differential Dijet Cross Section at the LHC*, Phys. Rev. Lett. **123**(10), 102001 (2019), doi:10.1103/PhysRevLett.123.102001, [1905.09047].

[92] X. Chen, T. Gehrmann, E. W. N. Glover, A. Huss and J. Mo, *NNLO QCD corrections in full colour for jet production observables at the LHC*, JHEP **09**, 025 (2022), doi:10.1007/JHEP09(2022)025, [2204.10173].

[93] X. Chen, T. Gehrmann, E. W. N. Glover and J. Mo, *Antenna subtraction for jet production observables in full colour at NNLO*, JHEP **10**, 040 (2022), doi:10.1007/JHEP10(2022)040, [2208.02115].

[94] F. A. Berends and W. T. Giele, *Recursive Calculations for Processes with n Gluons*, Nucl. Phys. B **306**, 759 (1988), doi:10.1016/0550-3213(88)90442-7.

[95] M. L. Mangano and S. J. Parke, *Multiparton amplitudes in gauge theories*, Phys. Rept. **200**, 301 (1991), doi:10.1016/0370-1573(91)90091-Y, [hep-th/0509223].

[96] J. Kuijf, *Multiparton production at hadron colliders*, Ph.D. thesis, Leiden (1991).

[97] Z. Bern, L. J. Dixon and D. A. Kosower, *One loop corrections to five gluon amplitudes*, Phys. Rev. Lett. **70**, 2677 (1993), doi:10.1103/PhysRevLett.70.2677, [hep-ph/9302280].

[98] Z. Bern, L. J. Dixon and D. A. Kosower, *One loop corrections to two quark three gluon amplitudes*, Nucl. Phys. B **437**, 259 (1995), doi:10.1016/0550-3213(94)00542-M, [hep-ph/9409393].

[99] Z. Kunszt, A. Signer and Z. Trocsanyi, *One loop radiative corrections to the helicity amplitudes of QCD processes involving four quarks and one gluon*, Phys. Lett. B **336**, 529 (1994), doi:10.1016/0370-2693(94)90568-1, [hep-ph/9405386].

[100] A. Signer, *Helicity method for next-to-leading order corrections in QCD*, Ph.D. thesis, ETH Zurich (1995).

[101] C. Anastasiou, E. W. N. Glover, C. Oleari and M. E. Tejeda-Yeomans, *Two loop QCD corrections to massless quark gluon scattering*, Nucl. Phys. B **605**, 486 (2001), doi:10.1016/S0550-3213(01)00195-X, [hep-ph/0101304].

[102] E. W. N. Glover, C. Oleari and M. E. Tejeda-Yeomans, *Two loop QCD corrections to gluon-gluon scattering*, Nucl. Phys. B **605**, 467 (2001), doi:10.1016/S0550-3213(01)00210-3, [hep-ph/0102201].

[103] E. W. N. Glover and M. E. Tejeda-Yeomans, *One loop QCD corrections to gluon-gluon scattering at NNLO*, JHEP **05**, 010 (2001), doi:10.1088/1126-6708/2001/05/010, [hep-ph/0104178].

[104] C. Anastasiou, E. W. N. Glover, C. Oleari and M. E. Tejeda-Yeomans, *Two-loop QCD corrections to the scattering of massless distinct quarks*, Nucl. Phys. B **601**, 318 (2001), doi:10.1016/S0550-3213(01)00079-7, [hep-ph/0010212].

[105] C. Anastasiou, E. W. N. Glover, C. Oleari and M. E. Tejeda-Yeomans, *Two loop QCD corrections to massless identical quark scattering*, Nucl. Phys. B **601**, 341 (2001), doi:10.1016/S0550-3213(01)00080-3, [hep-ph/0011094].

[106] C. Anastasiou, E. W. N. Glover, C. Oleari and M. E. Tejeda-Yeomans, *One loop QCD corrections to quark scattering at NNLO*, Phys. Lett. B **506**, 59 (2001), doi:10.1016/S0370-2693(01)00356-2, [hep-ph/0012007].

[107] J. Currie, A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and J. Pires, *Infrared sensitivity of single jet inclusive production at hadron colliders*, JHEP **10**, 155 (2018), doi:10.1007/JHEP10(2018)155, [1807.03692].

[108] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and T. A. Morgan, *Precise QCD predictions for the production of a Z boson in association with a hadronic jet*, Phys. Rev. Lett. **117**(2), 022001 (2016), doi:10.1103/PhysRevLett.117.022001, [1507.02850].

[109] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and T. A. Morgan, *The NNLO QCD corrections to Z boson production at large transverse momentum*, JHEP **07**, 133 (2016), doi:10.1007/JHEP07(2016)133, [1605.04295].

[110] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and T. A. Morgan, *NNLO QCD corrections for Drell-Yan p_T^Z and ϕ^* observables at the LHC*, JHEP **11**, 094 (2016), doi:10.1007/JHEP11(2016)094, [Erratum: JHEP 10, 126 (2018)], [1610.01843].

[111] R. Gauld, A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover and A. Huss, *Precise predictions for the angular coefficients in Z-boson production at the LHC*, JHEP **11**, 003 (2017), doi:10.1007/JHEP11(2017)003, [1708.00008].

[112] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and D. M. Walker, *Vector Boson Production in Association with a Jet at Forward Rapidities*, Eur. Phys. J. C **79**(6), 526 (2019), doi:10.1140/epjc/s10052-019-7010-2, [1901.11041].

[113] R. Gauld, A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss, I. Majer and A. Rodriguez Garcia, *Transverse momentum distributions in low-mass Drell-Yan lepton pair production at NNLO QCD*, Phys. Lett. B **829**, 137111 (2022), doi:10.1016/j.physletb.2022.137111, [2110.15839].

[114] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss, C. T. Preuss and D. M. Walker, *Precision phenomenology with fiducial cross sections in the triple-differential Drell-Yan process*, JHEP **05**, 002 (2023), doi:10.1007/JHEP05(2023)002, [2301.11827].

[115] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover, A. Huss and D. M. Walker, *Next-to-Next-to-Leading-Order QCD Corrections to the Transverse Momentum Distribution of Weak Gauge Bosons*, Phys. Rev. Lett. **120**(12), 122001 (2018), doi:10.1103/PhysRevLett.120.122001, [1712.07543].

[116] S. Frixione, *Isolated photons in perturbative QCD*, Phys. Lett. B **429**, 369 (1998), doi:10.1016/S0370-2693(98)00454-7, [hep-ph/9801442].

[117] K. Koller, T. F. Walsh and P. M. Zerwas, *Testing QCD: Direct Photons in e^+e^- Collisions*, Z. Phys. C **2**, 197 (1979), doi:10.1007/BF01474661.

[118] X. Chen, T. Gehrmann, N. Glover, M. Höfer and A. Huss, *Isolated photon and photon+jet production at NNLO QCD accuracy*, JHEP **04**, 166 (2020), doi:10.1007/JHEP04(2020)166, [1904.01044].

[119] X. Chen, T. Gehrmann, E. W. N. Glover, M. Höfer, A. Huss and R. Schürmann, *Single photon production at hadron colliders at NNLO QCD with realistic photon isolation*, JHEP **08**, 094 (2022), doi:10.1007/JHEP08(2022)094, [2205.01516].

[120] L. Bourhis, M. Fontannaz and J. P. Guillet, *Quarks and gluon fragmentation functions into photons*, Eur. Phys. J. C **2**, 529 (1998), doi:10.1007/s100520050158, [hep-ph/9704447].

[121] T. Gehrmann, E. W. N. Glover, A. Huss and J. Whitehead, *Scale and isolation sensitivity of diphoton distributions at the LHC*, JHEP **01**, 108 (2021), doi:10.1007/JHEP01(2021)108, [2009.11310].

[122] V. Del Duca, W. B. Kilgore and F. Maltoni, *Multiphoton amplitudes for next-to-leading order QCD*, Nucl. Phys. B **566**, 252 (2000), doi:10.1016/S0550-3213(99)00663-X, [hep-ph/9910253].

[123] A. Signer, *One loop corrections to five parton amplitudes with external photons*, Phys. Lett. B **357**, 204 (1995), doi:10.1016/0370-2693(95)00905-Z, [hep-ph/9507442].

[124] C. Anastasiou, E. W. N. Glover and M. E. Tejeda-Yeomans, *Two loop QED and QCD corrections to massless fermion boson scattering*, Nucl. Phys. B **629**, 255 (2002), doi:10.1016/S0550-3213(02)00140-2, [hep-ph/0201274].

[125] J. M. Campbell and C. Williams, *Triphoton production at hadron colliders*, Phys. Rev. D **89**(11), 113001 (2014), doi:10.1103/PhysRevD.89.113001, [1403.2641].

[126] X. Chen, T. Gehrmann, E. W. N. Glover and M. Jaquier, *Precise QCD predictions for the production of Higgs + jet final states*, Phys. Lett. B **740**, 147 (2015), doi:10.1016/j.physletb.2014.11.021, [1408.5325].

[127] M. A. Shifman, A. I. Vainshtein and V. I. Zakharov, *Remarks on Higgs Boson Interactions with Nucleons*, Phys. Lett. B **78**, 443 (1978), doi:10.1016/0370-2693(78)90481-1.

[128] F. Wilczek, *Decays of Heavy Vector Mesons Into Higgs Particles*, Phys. Rev. Lett. **39**, 1304 (1977), doi:10.1103/PhysRevLett.39.1304.

[129] V. Del Duca, A. Frizzo and F. Maltoni, *Higgs boson production in association with three jets*, JHEP **05**, 064 (2004), doi:10.1088/1126-6708/2004/05/064, [hep-ph/0404013].

[130] L. J. Dixon and Y. Sofianatos, *Analytic one-loop amplitudes for a Higgs boson plus four partons*, JHEP **08**, 058 (2009), doi:10.1088/1126-6708/2009/08/058, [0906.0008].

[131] S. Badger, E. W. N. Glover, P. Mastrolia and C. Williams, *One-loop Higgs plus four gluon amplitudes: Full analytic results*, JHEP **01**, 036 (2010), doi:10.1007/JHEP01(2010)036, [0909.4475].

[132] S. Badger, J. M. Campbell, R. K. Ellis and C. Williams, *Analytic results for the one-loop NMHV Hqqgg amplitude*, JHEP **12**, 035 (2009), doi:10.1088/1126-6708/2009/12/035, [0910.4481].

[133] T. Gehrmann, M. Jaquier, E. W. N. Glover and A. Koukoutsakis, *Two-Loop QCD Corrections to the Helicity Amplitudes for $H \to 3$ partons*, JHEP **02**, 056 (2012), doi:10.1007/JHEP02(2012)056, [1112.3554].

[134] X. Chen, J. Cruz-Martinez, T. Gehrmann, E. W. N. Glover and M. Jaquier, *NNLO QCD corrections to Higgs boson production at large transverse momentum*, JHEP **10**, 066 (2016), doi:10.1007/JHEP10(2016)066, [1607.08817].

[135] X. Chen, T. Gehrmann, E. W. N. Glover and A. Huss, *Fiducial cross sections for the four-lepton decay mode in Higgs-plus-jet production up to NNLO QCD*, JHEP **07**, 052 (2019), doi:10.1007/JHEP07(2019)052, [1905.13738].

[136] X. Chen, T. Gehrmann, E. W. N. Glover and A. Huss, *Fiducial cross sections for the lepton-pair-plus-photon decay mode in Higgs production up to NNLO QCD*, JHEP **01**, 053 (2022), doi:10.1007/JHEP01(2022)053, [2111.02157].

[137] M. Cacciari, G. P. Salam and G. Soyez, *The anti-k_t jet clustering algorithm*, JHEP **04**, 063 (2008), doi:10.1088/1126-6708/2008/04/063, [0802.1189].

[138] M. Wobisch and T. Wengler, *Hadronization corrections to jet cross-sections in deep inelastic scattering*, In *Workshop on Monte Carlo Generators for HERA Physics (Plenary Starting Meeting)*, pp. 270–279 (1998), [hep-ph/9907280].

[139] S. D. Ellis and D. E. Soper, *Successive combination jet algorithm for hadron collisions*, Phys. Rev. D **48**, 3160 (1993), doi:10.1103/PhysRevD.48.3160, [hep-ph/9305266].

[140] S. Catani, Y. L. Dokshitzer, M. H. Seymour and B. R. Webber, *Longitudinally invariant K_t clustering algorithms for hadron hadron collisions*, Nucl. Phys. B **406**, 187 (1993), doi:10.1016/0550-3213(93)90166-M.

[141] S. Catani, Y. L. Dokshitzer, M. Olsson, G. Turnock and B. R. Webber, *New clustering algorithm for multi - jet cross-sections in e+ e- annihilation*, Phys. Lett. B **269**, 432 (1991), doi:10.1016/0370-2693(91)90196-W.

[142] W. Bartel *et al.*, *Experimental Studies on Multi-Jet Production in e^+e^- Annihilation at PETRA Energies*, Z. Phys. C **33**, 23 (1986), doi:10.1007/BF01410449.

[143] F. Siegert, *A practical guide to event generation for prompt photon production with Sherpa*, J. Phys. G **44**(4), 044007 (2017), doi:10.1088/1361-6471/aa5f29, [1611.07226].

[144] D. Buskulic *et al.*, *First measurement of the quark to photon fragmentation function*, Z. Phys. C **69**, 365 (1996), doi:10.1007/BF02907417.

[145] M. Tanabashi *et al.*, *Review of Particle Physics*, Phys. Rev. D **98**(3), 030001 (2018), doi:10.1103/PhysRevD.98.030001.

[146] P. A. Zyla *et al.*, *Review of Particle Physics*, PTEP **2020**(8), 083C01 (2020), doi:10.1093/ptep/ptaa104.

[147] A. Gehrmann-De Ridder, T. Gehrmann, N. Glover, A. Huss and T. A. Morgan, *NNLO QCD corrections for Z boson plus jet production*, PoS **RADCOR2015**, 075 (2016), doi:10.22323/1.235.0075, [1601.04569].