PUBLISHED BY INSTITUTE OF PHYSICS PUBLISHING FOR SISSA/ISAS

RECEIVED: *May 26, 2003*
REVISED: *June 30, 2003*
ACCEPTED: *July 1, 2003*

ALPGEN, a generator for hard multiparton processes in hadronic collisions

Michelangelo L. Mangano, Fulvio Piccinini* and Antonio D. Polosa

CERN, Theoretical Physics Division
CH 1211 Geneva 23, Switzerland
E-mail: michelangelo.mangano@cern.ch, fulvio.piccinini@cern.ch, antonio.polosa@cern.ch

Mauro Moretti

Dipartimento di Fisica, Università di Ferrara, and INFN
Ferrara, Italy
E-mail: mauro.moretti@fe.infn.it

Roberto Pittau

Dipartimento di Fisica, Università di Torino, and INFN
Torino, Italy
E-mail: roberto.pittau@to.infn.it

ABSTRACT: This paper presents a new event generator, ALPGEN, dedicated to the study of multiparton hard processes in hadronic collisions. The code performs, at the leading order in QCD and EW interactions, the calculation of the exact matrix elements for a large set of parton-level processes of interest in the study of the Tevatron and LHC data. The current version of the code describes the following final states: $(W \to f\bar{f}')Q\overline{Q}+N$ jets (Q being a heavy quark, and $f=\ell,q$), with $N\leq 4$; $(Z/\gamma^* \to f\bar{f})\,Q\overline{Q}+N$ jets ($f=\ell,\nu$), with $N\leq 4$; $(W\to f\bar{f}')$ + charm + N jets ($f=\ell,q$, $N\leq 5$); $(W\to f\bar{f}')+N$ jets ($f=\ell,q$) and $(Z/\gamma^*\to f\bar{f})+N$ jets ($f=\ell,\nu$), with $N\leq 6$; $nW+mZ+lH+N$ jets, with $n+m+l+N\leq 8$, $N\leq 3$, including all 2-fermion decay modes of W and Z bosons, with spin correlations; $Q\overline{Q}+N$ jets, with $t\to bf\bar{f}'$ decays and relative spin correlations included where relevant, and $N\leq 6$; $Q\overline{Q}Q'\overline{Q}'+N$ jets, with Q and Q' heavy quarks (possibly equal) and $N\leq 4$; $HQ\overline{Q}+N$ jets, with $t\to bf\bar{f}'$ decays and relative spin correlations included where relevant and $N\leq 4$; N jets, with $N\leq 6$. Parton-level events are generated, providing full information on their colour and flavour structure, enabling the evolution of the partons into fully hadronised final states.

KEYWORDS: QCD, Jets, Hadronic Colliders.

*On leave of absence from INFN Sezione di Pavia, Italy.

JHEP07(2003)001

Contents

1. Introduction

The large energies available in current and forthcoming hadronic colliders make final states with several hard and well separated jets a rather common phenomenon. These multijet final states can originate directly from hard QCD radiative processes [1], or from the decay of massive particles, such as for example W and Z gauge bosons, top quarks, Higgs bosons, supersymmetric particles, etc. Whether our interest is in accurate measurements of top quarks or in the search for more exotic states [2], multijet final states always provide an important observable, and the study of the backgrounds due to QCD is an essential part of any experimental analysis.

Several examples of calculation of multijet cross-sections in hadronic collisions exist in the literature. Some of them are included in parton-level Monte Carlo (MC) event generators, where final states consisting of hard and well isolated partons are generated. Among the most used and best documented examples are PAPAGENO [3] (a compilation of several partonic processes), VECBOS [4] (for production of W/Z bosons in association with up to 4 jets), NJETS [5] (for production of up to 6 jets). The range of jet multiplicities calculable in practice for purely QCD processes in hadronic collisions has recently been extended in [6], where new techniques [7] to deal with the complexity of the multijet flavour configurations have been implemented in the calculation of up to 7 jets. Finally, programs for the automatic generation of user-specified parton-level processes exist and have been used in the literature for the calculation of many important reactions in hadronic collisions: MADGRAPH [8], CompHEP [9], GRACE [10] and AMEGIC++ [11].

In order to use these results in practical analyses of the experimental data, the calculations need to be completed with the treatment of the higher-order corrections leading to the development of partonic cascades, and with the subsequent transformation of the partons into observable hadrons. MC programs such as HERWIG [12], PYTHIA [13] or ISAJET [14] are available to carry out these last two steps. The consistent combination of the parton-level calculations with the partonic evolution given by the shower MC programs is the subject of extensive work. Several approaches to this problem have been proposed in the case of low-order processes, where one is interested in final states with at most one extra jet relative to a given Born-level configuration (for example W, Z plus jet [15] and $t\bar{t}$ + jet [16]). In these cases, the proposed algorithms correct the probability for hard-emission estimated by the approximation of the shower-evolution programs, using the value of the exact real-emission higher-order matrix element. In other studies [17, 18, 19], limited for the time being to the case of jet emission in association with vector boson and heavy quark [20] pairs, the full set of virtual and real NLO corrections to the partonic matrix elements has been merged with the HERWIGand PYTHIAMC's. See in particular [19] for a complete discussion of the problem of matching NLO parton level and leading-logarithmic (LL) shower generators. Some of the problems raised by a consistent matching of NLO parton level calculations and next-to-leading logarithm (NLL) shower evolution are discussed in [21].

In the case of large jet multiplicities, the complexity of the matrix element evaluation and of its singularity structure prevents so far the application of these approaches. Recently, a new strategy has been introduced [22] for the merging of multijet matrix elements

JHEP07(2003)001

with the shower development. This involves a reweighting of the matrix element weights with Sudakov form factors, and the veto of shower emissions in regions of phase-space already covered by the parton-level configurations. After the shower evolution, samples of different parton-level multiplicity can then be combined together to obtain inclusive samples of arbitrary jet multiplicity, double counting being limited to subleading effects. A complete application of this ideas has been achieved for inclusive hadronic final states in e^+e^- collisions, and a proposal has been formulated for the extension to the hadron collider case [23]. Explicit implementations are being developed [24].

We discussed in [25] a theoretical framework for the evaluation and generation of events in such a way as to enable the subsequent perturbative evolution using a shower MC program. In [26] we presented a complete application to the case where a W boson is produced in association with a heavy quark pair, plus up to 4 additional light partons. The relative MC code allows a complete description of these final states, from the leading order (LO) matrix element computation to the perturbative evolution and hadronization carried out using HERWIG.

We recently extended the application of the ideas contained in [25], completing a library of MC codes for hadronic collisions including the following new processes: $WQ\overline{Q}+N$ jets and $Z/\gamma^*\, Q\overline{Q}+N$ jets (Q being a heavy quark), with $N\leq 4$; $W+$ charm $+\,N$ jets; $W+N$ jets and $Z+N$ jets ($N\leq 6$); $nW+mZ+lH+N$ jets, with $n+m+l+N\leq 8$ and $N\leq 3$; $Q\overline{Q}+N$ jets ($N\leq 6$); $Q\overline{Q}Q'\overline{Q}'+N$ jets, with Q and Q' heavy quarks (possibly equal) and $N\leq 4$; $HQ\overline{Q}+N$ jets ($N\leq 4$); N jets, with $N\leq 6$. Details on the decay mode options for the various unstable particles will be given below. For all of these processes, an interface to both HERWIG and PYTHIA is provided. In the present work, we document the contents and use of this library. The emphasis is on the code itself and not on the physics approach, which is discussed in more detail in [26]. Numerical results for some benchmark processes are nevertheless presented.

Independent work on the merging of parton-level calculations with shower MC's (for the specific case of PYTHIA) has been pursued by the CompHEP group [27], by the GRACE group [28] and, more recently, in [29, 30, 31]. To the best of our knowledge, a large fraction of the matrix element calculations documented in this paper have never been performed before.

Section 2 reviews the general structure of the codes, covering both the aspects of the parton-level calculations, and of the shower evolution. Section 3 discusses the features of the implementation of each individual hard process in our library. A technical appendix will provide more explicit details on the programs and their use.

The library containing all codes described in this work can be downloaded from the following URL: `http://mlm.home.cern.ch/mlm/alpgen`.

2. The general structure of the program

The program consists of several building blocks (see the appendix for details). A section of the code library defines the overall infrastructure of the generator, implementing the logical sequence of operations in a standard set of subroutine calls; this part is independent of the

JHEP07(2003)001

hard process selected and, among other things, it includes the algorithms needed for the evaluation of the matrix elements, for the evaluation of the parton densities and for the bookkeeping and histogramming of the results. Each hard process has viceversa a separate set of code elements, which are specific to it. These include the process initialization, the phase-space generation, the extraction of flavour and colour structure of the event, and the default analysis routines. Each hard process corresponds to a specific executable, obtained by linking the relative process-dependent code elements with the process-independent ones. In addition to the above, a section of the code library deals with the shower evolution. As explained below, the shower evolution is performed as an independent step, following the generation of a sample of unweighted parton-level events. The code elements relative to this phase of the computation include the HERWIG code and the algorithm needed to transform the partonic input into a format which can be interpreted and processed by HERWIG. We adopt the formatting standard proposed in the so called Les Houches accord [32].

As alluded to above, the program has two main modes of operation. In the first mode the code performs the parton-level calculation of the matrix elements relative to the selected hard processes, generating weighted events. Each weighted event is analysed on-line in a routine where the kinematics of the event can be studied, and histograms filled. The user has direct access to this analysis routine, and can adapt it to his needs. At the end of the run, differential distributions are obtained. A histogramming package is included in the code library, which generates a graphic output in the form of topdrawer plots. This mode of running can also be used in order to easily get total cross sections in presence of some overall generation cuts (e.g. rates for production of jets above a given threshold), without looking at any particular differential distribution.

In the second mode of operation the code generates unweighted parton level events and stores them to a file, for subsequent evolution via the parton-shower part of the program. The generation of parton-level events, and their shower evolution, are performed in two different phases by different code elements. Since the generation of unweighted parton-level events is typically by far the most CPU intensive component of the calculation, the storage of unweighted events allows to build up event data sets which can then be used efficiently for studies of hadronization systematics or realistic detector simulations. In this mode of operation the matrix-element calculation generates all the flavour and colour information necessary for the complete shower evolution. The kinematical, flavour and colour data for a given event are stored in a file, and are read in by the shower MC, which will process the event. In the rest of this section we present in more detail these two running modes of the code.

2.1 Parton-level generation and cross section evaluation

In a nutshell, the calculation of the cross section for a given hard process is performed in the following steps:

- The parameters required to define the hard process are passed to the code. These include the selection of jet multiplicity, the mass of possible heavy quarks, rapidity and transverse momentum cuts, etc.

JHEP07(2003)001

- A first phase-space integration cycle is performed, with the goal of exploring how the cross-section is distributed in phase-space and among the possible contributing subprocesses. Event by event, the following steps take place:
  - one subprocess (see later) is randomly selected;
  - a point in phase-space is randomly selected, consistent with the required kinematical acceptance cuts;
  - the initial-state parton luminosity is evaluated for the chosen subprocess, and one among the possible flavour configurations is selected (see later);
  - spin and colour for each parton are randomly assigned;
  - the matrix element is evaluated, and the weight of the event is obtained after inclusion of the phase-space and parton-luminosity factors. A bookkeeping of the weights is kept for each individual subprocess and phase-space subvolume.
- At the end of the first integration iteration, a map of the cross-section distribution among the different subprocesses and in phase-space is available. It will be used for subsequent integration cycles, where the phase-space and subprocess random sampling will be weighted by the respective probability distributions.
- After the completion of a series of warm-up integration cycles (whose number is specified at the beginning of the run by the user), the optimised integration grids are stored in a file. A final large-statistics run is then performed. After the generation of each event, its kinematics is analysed and histograms are filled.

We shall now discuss in more detail the individual steps outlined above and the ingredients of the calculation.

2.1.1 Selection of the subprocess

The calculation of the cross section for multiparton final states involves typically the sum over a large set of subprocesses and flavour configurations. For example, in the case of $WQ\overline{Q}+2$ jets we have, among others, the following subprocesses:

$$q\bar{q}' \to WQ\overline{Q}gg,\ qg \to WQ\overline{Q}gq',\ gq \to WQ\overline{Q}gq',\ gg \to WQ\overline{Q}q\bar{q}',\ q\bar{q}' \to WQ\overline{Q}q''\bar{q}'',\ \text{etc.} \tag{2.1}$$

Each of these subprocesses receives contributions from several possible flavour configurations (e.g. $u\bar{d} \to WQ\overline{Q}gg$, $u\bar{s} \to WQ\overline{Q}gg$, etc.). Our subdivision in subprocesses is designed to allow to sum the contribution of different flavour configurations by simply adding trivial factors such as parton densities or CKM factors, which factorize out of a single, flavour independent, matrix element. For example the overall contribution from the first process in the above list is given by

$$\left[u_1\bar{d}_2\cos^2\theta_c + u_1\bar{s}_2\sin^2\theta_c + c_1\bar{s}_2\cos^2\theta_c + c_1\bar{d}_2\sin^2\theta_c\right] \times |M(q\bar{q}' \to WQ\overline{Q}gg)|^2\ , \tag{2.2}$$

where $q_i = f(x_i)$, $i = 1, 2$, are the parton densities for the quark flavour q. Contributions from charge-conjugate or isospin-rotated states can also be summed up, after trivial momentum exchanges. For example, the same matrix element calculation is used to describe the four events:

$$\begin{aligned}
u(p_1)\bar{d}(p_2) &\to b(p_3)\bar{b}(p_4)g(p_5)g(p_6)e^+(p_5)\nu(p_6)\\
\bar{u}(p_1)d(p_2) &\to \bar{b}(p_3)b(p_4)g(p_5)g(p_6)e^-(p_5)\bar{\nu}(p_6)\\
\bar{d}(p_1)u(p_2) &\to \bar{b}(p_3)b(p_4)g(p_5)g(p_6)\nu(p_5)e^+(p_6)\\
d(p_1)\bar{u}(p_2) &\to b(p_3)\bar{b}(p_4)g(p_5)g(p_6)\bar{\nu}(p_5)e^-(p_6)\,.
\end{aligned}$$

Event by event, the flavour configuration for the assigned subprocess is then selected with a probability proportional to the relative size of the individual contributions to the luminosity, weighted by the Cabibbo angles.

Typically, only few among all possible subprocesses give a substantial contribution to the cross section. For each event, instead of summing the weight of all subprocesses, we calculate only one. This is selected with uniform probability during the first integration cycle, and a record is kept of each individual contribution to the cross-section. In subsequent integration iterations, the accumulated rates of the single channels are used to weight their selection probabilities. This will significantly improve the CPU performance of the code, and the unweighting efficiency.

Due to the rapid growth in the number of subprocesses when quarks are added [7], we limited ourselves to processes with at most 2 pairs of light quarks (plus pairs of heavy quarks, when required). In all of the cases considered this is not, however, a significant limitation on the accuracy of the results. The full list of subprocesses available for each hard process is given in section 3.

2.1.2 Phase-space sampling

The phase-space generation is optimised for each individual hard process, using generation variables which are most suitable to the application of typical hadron collider selection cuts. The phase-space is mapped with a multidimensional grid, and during the integration a record is kept of the total weight accumulated within each bin of the grid. To further contribute to the efficiency of the phase-space sampling, independent grids are employed to sample different subprocesses. In particular, we shall associate one phase-space grid to each of the following initial states:

1. $q\bar{q}$, $q\bar{q}'$ and charge conjugates
2. qg and $\bar{q}g$
3. gq and $g\bar{q}$
4. gg
5. qq, qq' and charge conjugates.

For example, the processes $q\bar{q}' \to WQ\overline{Q}gg$ and $q\bar{q}' \to WQ\overline{Q}q\bar{q}$ share the same phase-space grid.

2.1.3 Matrix element calculation

The calculation of the LO matrix elements for the selected hard process is performed using the ALPHA [33] algorithm, extended to QCD interactions as described in [25, 26]. As explained in detail in [26], use of the ALPHA algorithm is in our view essential in order to cope with the complexity of the problem. All mass effects are included in the case of massive quarks. The calculations are done after having selected, on an event-by-event basis, polarization, flavour and colour configurations, in order to be able to provide the shower MC's with the information necessary for the shower evolution. The sum over polarizations and colours is performed by summing over multiple events, in a MC fashion. The choice of colour basis and the strategy for the determination of the colour flows necessary for the coherent shower evolution are described in [26]. To the best of our knowledge, a large fraction of the matrix element calculations documented in this paper have never been performed before using other calculational tools. An independent implementation of the ALPHA algorithm has been exploited for the evaluation of multijet processes in hadronic collisions in [34, 6].

2.1.4 PDF sets and α_s

The code library includes a choice among some of the most recent PDF parameterizations. They can be selected at the beginning of the run through the variable `ndns`, which is mapped as follows:[1]

`ndns`	PDF	$[\alpha_s(m_Z)]_{n_{\rm loop}}$	`ndns`	PDF	$[\alpha_s(m_Z)]_{n_{\rm loop}}$
1	CTEQ4M [35]	$[0.116]_2$	101	MRST99-1 [38]	$[0.1175]_2$
2	CTEQ4L [35]	$[0.116]_2$	102	MRST01-1 [39]	$[0.119]_2$
3	CTEQ4HJ [35]	$[0.116]_2$	103	MRST01-2 [39]	$[0.117]_2$
4	CTEQ5M [36]	$[0.118]_2$	104	MRST01-3 [39]	$[0.121]_2$
5	CTEQ5L [36]	$[0.127]_1$	105	MRST01J [39]	$[0.121]_2$
6	CTEQ5HJ [36]	$[0.118]_2$	106	MRSTLO [40]	$[0.130]_1$
7	CTEQ6M [37]	$[0.118]_2$			
8	CTEQ6L [37]	$[0.118]_2$			

In the case of NLO sets we use the 2-loop expression for α_s:

$$\alpha_s(Q) = \frac{1}{b_5\ \log(Q^2/\Lambda_5^2)} - \frac{b'_5}{(b_5\ \log^2(Q^2/\Lambda_5^2))\log\log(Q^2/\Lambda_5^2)}\,. \tag{2.3}$$

valid for $Q > m_b \equiv 4.5\,\text{GeV}$, where b_5 and b'_5 are the 1- and 2-loop coefficients of the QCD β function, respectively, for 5 flavours. Threshold matching is applied in the case of $Q < m_b$. In the case of LO sets (such as CTEQ*L or MRSTLO) we follow the prescriptions used by the authors in performing the PDF fits. These vary from set to set. For example, set CTEQ5L was fitted using a LO expression for α_s, while CTEQ6L used the NLO

[1] In addition to these default sets, we include in the package the full group of 40 CTEQ6M sets which allow the determination of PDF systematics errors [37]. To access these, the file `alplib/alppdf.f` should be replaced in the linking stage with the file `alplib/alppdf_cteq.f`. A more complete set of parton densities, including old sets back to EHLQ and DO, is also available upon request.

evolution. The table above lists the values of $\alpha_s(m_Z)$ corresponding to the various sets, and indicates whether these values (and the relative evolution to different renormalization scales) correspond to the 1 or 2 loop formulation.

2.1.5 Electroweak couplings

In the current version of the ALPHA code the input couplings are derived from the standard $\mathrm{SU}(3) \times \mathrm{SU}(2)_L \times \mathrm{U}(1)_Y$ tree level lagrangian. The choice of input EW parameters deserves a short discussion. In ALPHA, the couplings of the electroweak (EW) and Higgs (H) bosons (including the respective selfcouplings) are parametrised in terms of the SU(2) coupling strength g, of the weak mixing angle $\sin\theta_W$, of the electromagnetic fine structure constant α_{em}, and of the masses of W, Z and H. We therefore have a total of 6 parameters needed to specify the value of general EW matrix elements. If we want to preserve gauge invariance we are however allowed to use only four independent parameters (plus fermion masses in the Yukawa sector). Treating m_H as a free parameter, gauge invariance at the tree level demands that the remaining 5 parameters satisfy the following tree-level relationships:

$$\cos\theta_W = \frac{m_W}{m_Z} \tag{2.4}$$

$$e = g\ \sin\theta_W\,. \tag{2.5}$$

The Higgs self-couplings and Yukawa couplings to fermions are furthermore given by:

$$\lambda_{hhh} = \frac{g\, m_H^2}{4m_W} \tag{2.6}$$

$$\lambda_{hhhh} = \frac{g^2\, m_H^2}{32\, M_W^2} \tag{2.7}$$

$$y_f = \frac{g m_f}{\sqrt{2} m_W}\,. \tag{2.8}$$

As a consequence, we cannot assign to the input parameters the values which are known from the current accurate experimental measurements, since these values are only consistent with the radiatively corrected versions of the above relations. This is not a merely formal issue: any tiny violation of the tree-level gauge relationships among the model parameters leads to violations of the equivalence theorem and leads to unphysical corrections to the tree-level cross-sections scaling like $(E_V/M_V)^{2n}$, n being the number of on/off-shell heavy gauge bosons appearing in the relevant diagrams and E_V, M_V their energy and mass respectively. In view of the large center of mass energy available in current and future hadron collisions, these spurious corrections could be numerically large, leading to the wrong high-energy behaviour of the cross sections.

The current version of the code provides four choices for the setting of EW parameters. These choices are controlled by the variable `iewopt`, set at running time (default values for this variable are provided in the code, and are listed in the following sections describing the individual hard processes). The different options are listed here below; the numerical values of the calculated parameters are obtained using the following set of inputs: $m_W = 80.41$,

$m_Z = 91.188$, $\sin^2\theta_W = 0.231$, $\alpha_{em}(m_Z) = 1/128.89$, $G_F = 1.16639 \times 10^{-5}$:

`iewopt=0`. Inputs: $\alpha_{em}(m_Z)$, G_F, $\sin^2\theta_W$. Extracted:

$$
\begin{aligned}
g &= \frac{\sqrt{4\pi\alpha_{em}(m_Z)}}{\sin\theta_W} = 0.6497\,, \qquad m_W = \frac{g}{\sqrt{4\sqrt{2}G_F}} = 79.98\,,\\
m_Z &= \frac{m_W}{\cos\theta_W} = 91.20
\end{aligned}
\tag{2.9}
$$

`iewopt=1`. Inputs: m_W, G_F, $\sin^2\theta_W$. Extracted:

$$
\begin{aligned}
m_Z &= \frac{m_W}{\cos\theta_W} = 91.695\,, \qquad g = \left(4\sqrt{2}\,G_F\right)^{1/2} m_W = 0.6532\,,\\
\alpha_{em}(m_Z) &= \frac{(g\sin\theta_W)^2}{4\pi} = \frac{1}{127.51}
\end{aligned}
\tag{2.10}
$$

`iewopt=2`. Inputs: m_Z, $\alpha_{em}(m_Z)$, $\sin^2\theta_W$. Extracted:

$$
m_W = m_Z\cos\theta_W = 79.97\,, \qquad g = \frac{\sqrt{4\pi\alpha_{em}(m_Z)}}{\sin\theta_W} = 0.6497
\tag{2.11}
$$

`iewopt=3`. Inputs: m_Z, m_W, G_F. Extracted:

$$
\begin{aligned}
\sin^2\theta_W &= 1 - \left(\frac{m_W}{m_Z}\right)^2 = 0.2224\,, \qquad g = \left(4\sqrt{2}\,G_F\right)^{1/2} m_W = 0.653\,,\\
\alpha_{em}(m_Z) &= \frac{(g\ \sin\theta_W)^2}{4\pi} = \frac{1}{132.42}\,.
\end{aligned}
\tag{2.12}
$$

As a default, in all processes we employ `iewopt=3`. We verified that alternative options generate changes in the rates by at most few percent. Gauge and Higgs boson widths are calculated at tree level after the couplings have been selected. With the exception of the class of processes involving several gauge bosons, which will be discussed in detail section 3.6 and where we set boson widths to 0, ALPHA uses fixed widths in the propagators.

2.2 Unweighting and Shower evolution

The starting point for the processing of events through the shower evolution is the generation of a sample of unweighted events. This generation takes place through a two-step procedure (more details are given in the appendix):

1. to start with, a run of the parton-level code is performed as described in section 2.1. Selecting the running mode option `imode=1`, weighted events are stored in a file. To limit the size of the file, instead of saving all the event information, we simply store the seed of the first random number used in the generation of the event, in addition to the event weight.

2. at the end of the generation of the weighted event sample, the unweighting is performed by running the code once more, using a running mode option `imode=2`. In this running mode, the code will sequentially read the events stored in the file, and will perform the unweighting using the knowledge of the maximum weight of the sample, and of the weight of each individual event. When an event is selected by the unweighting, the seed of the random number is uploaded and all the information about the event (kinematics, flavours, spins and colours) is automatically reconstructed. Starting from the stored seed the event kinematics is reproduced exactly (on the same architecture) or up to machine accuracy (a few 10^{-16}) on any other platform. A check on the reconstructed event is made by comparing the generated `x1` value (`imode=2`) with the one generated using `imode=1` and stored in the file `file.wgt`. If the two values are found to be different, the program outputs an error message and stops. The colour flow for the event is then selected according to the algorithm described in [26]: the subamplitudes corresponding to all colour flows compatible with the colour state of the event are first evaluated using ALPHA; one of them is then randomly extracted with a probability proportional to the squared modulus of the relative subamplitude. The momenta of the particles, together with their flavour and with the colour flow information, are then written to a file, which at the end of the run will contain the complete sample of unweighted events.

At this point we are ready to process the events through the shower evolution. The stored events can be read by the chosen shower MC, the kinematics, colour and flavour information for each event being translated into the event format established by the Les Houches convention [32].

3. The available hard processes

3.1 $\mathbf{WQ\overline{Q}}+$ jets

The physics content of the $WQ\overline{Q}+$ jets code has been described in detail in [26], where some phenomenological applications are also presented. We use the notation W as a short hand; what is actually calculated is the matrix element for a fermion-antifermion final state. All spin correlations and finite width effects are therefore accounted for. The quoted cross sections refer to a single lepton family; in the flavour assignment, the code selects by default an electron. Different flavours can be selected during the unweighting phase, covering all possible leptonic decays, as well as inclusive quark decays (for more details see the appendix B.4). In the case $Q = t$, the top quark is left undecayed. The EW parameters are fixed by default using the option `iewopt=3` (see eq. (2.10)). Only the leading-order EW diagrams are included in the calculation.

The subprocesses considered include all configurations with up to 2 light-quark pairs; they are listed in table 1, following the notation employed in the code.

For all processes, the charge-conjugate ones are always understood. The above list fully covers all the possible processes with up to 3 light jets in addition to the heavy quarks. In

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$q\bar{q}' \to WQ\overline{Q}$	2	$qg \to q'WQ\overline{Q}$	3	$gq \to q'WQ\overline{Q}$
4	$gg \to q\bar{q}'WQ\overline{Q}$	5	$q\bar{q}' \to WQ\overline{Q}q''\bar{q}''$	6	$qq'' \to WQ\overline{Q}q'q''$
7	$q''q \to WQ\overline{Q}q'q''$	8	$q\bar{q} \to WQ\overline{Q}q'\bar{q}''$	9	$q\bar{q}' \to WQ\overline{Q}q\bar{q}$
10	$\bar{q}'q \to WQ\overline{Q}q\bar{q}$	11	$q\bar{q} \to WQ\overline{Q}q\bar{q}'$	12	$q\bar{q} \to WQ\overline{Q}q'\bar{q}$
13	$qq \to WQ\overline{Q}qq'$	14	$qq' \to WQ\overline{Q}qq$	15	$qq' \to WQ\overline{Q}q'q'$
16	$gg \to WQ\overline{Q}q'q''\bar{q}''$	17	$gq \to WQ\overline{Q}q'q''\bar{q}''$	18	$qg \to WQ\overline{Q}qq\bar{q}'$
19	$gq \to WQ\overline{Q}q'q\bar{q}$	20	$gq \to WQ\overline{Q}qq\bar{q}'$	21	$gq \to WQ\overline{Q}q'q\bar{q}$
22	$gg \to WQ\overline{Q}q\bar{q}'q''\bar{q}''$	23	$gg \to WQ\overline{Q}q\bar{q}q\bar{q}'$		

Table 1: Subprocesses included in the $WQ\overline{Q}$+jets code. Additional final-state gluons are not explicitly shown here but are included in the code if the requested light-jet multiplicity ($N \leq 4$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1` in the case of 2 light jets will correspond to the final state $q\bar{q}' \to WQ\overline{Q}gg$. The details can be found in the subroutine `selflav` of the file `wqqlib/wqq.f`.

the case of 4 extra jets, we do not calculate processes with 3 light-quark pairs. Within the uncertainties of the LO approximation, these can be safely neglected [4].

As a default, the code generates kinematical configurations defined by cuts applied to the following variables (the cuts related to the heavy quarks are only applied in the case of b, while top quarks are always generated without cuts):

- ${p_T}^{\rm jet}$, $\eta^{\rm jet}$, ΔR_{jj}
- ${p_T}^b$, η^b, $\Delta R_{b\bar{b}}$
- ${p_T}^\ell$, η^ℓ, ${p_T}^\nu$, $\Delta R_{\ell j}$,

where $\Delta R_{ab} = \sqrt{[(\eta_a - \eta_b)^2 + (\phi_a - \phi_b)^2]}$. The cut values can be provided by the user at run time. Additional cuts can be supplied by the user in the routine `usrcut` contained in the user file `wqqwork/wqqusr.f`.

In the code initialization phase, the user can select among 3 continuous choices for the parametrization of the factorization and renormalization scale Q: a real input parameter (`qfac`) allows to vary the overall scale of Q, $Q = \texttt{qfac} \times Q_0$, while the preferred functional form for Q_0 is selected through the integer input parameter `iqopt`:

iqopt	0	1	2
Q_0^2	$m_W^2 + {p_T}_W^2$	m_W^2	$m_W^2 + \sum m_T^2$

where m_T is the transverse mass defined as $m_T^2 = m^2 + p_T^2$, and the sum $\sum m_T^2$ extends to all final state partons (including the heavy quarks, excluding the W decay products).

Some numerical benchmark results are given in table 2 and 3. In the case of the minimal jet multiplicity, the results agree with previous calculations (see e.g. [41, 42]). The following scale and cuts are used:

$$Q^2 = m_W^2 + p_{T,W}^2, \tag{3.1}$$

$${p_T}^{\rm jet} > 20\,\mathrm{GeV}, \quad |\eta_j| < 2.5, \quad \Delta R_{jj} > 0.7 \tag{3.2}$$

$${p_T}^b > 20\,\mathrm{GeV}, \quad |\eta_b| < 2.5, \quad \Delta R_{b\bar{b}} > 0.7, \Delta R_{bj} > 0.7\ . \tag{3.3}$$

JHEP07(2003)001

	$N=0$	$N=1$	$N=2$	$N=3$	$N=4$
LHC (pb)	2.222(4)	3.013(9)	1.83(1)	0.831(8)	0.307(5)
FNAL (fb)	332.2(7)	86.2(4)	18.3(2)	3.17(3)	0.44(3)

Table 2: $\sigma(b\bar{b}\ell\nu + N \text{ jets})$ at the Tevatron and at the LHC. Parameters and cuts are given in eqs. (3.1)–(3.3).

LHC (fb)	61.1(4)
FNAL (fb)	1.55(1)

Table 3: $\sigma(t\bar{t}\ell\nu)$ at the Tevatron and at the LHC.

Here and in the following sections we shall use the PDF set CTEQ5L. The quoted errors reflect the statistical accuracy of the integrations. We never tried to go beyond the percent level, to concentrate the CPU resources on the most computationally demanding channels. Results for the FNAL Tevatron refer to $p\bar{p}$ collisions at $\sqrt{S} = 2\,\text{TeV}$, those for the LHC refer to pp collisions at $\sqrt{S} = 14\,\text{TeV}$.

3.2 $\mathbf{Z/\gamma^* \, Q\overline{Q}}$+ jets

We use the notation Z/γ^* as a short hand; what is actually calculated is the matrix element for a lepton-pair final state. All spin correlations and finite width effects are therefore accounted for. When the final state $\ell^+\ell^-$ is selected, the interference between intermediate Z and γ^* is also included. The quoted cross sections refer to a single lepton family; in the flavour assignement, the code selects by default the e^+e^- pair. In the case of the final state $\nu\bar{\nu}$ the quoted cross sections include the decays to all 3 neutrino flavours, although we always label the neutrinos as ν_e. In the case $Q = t$, the top quark is left undecayed. The EW parameters are fixed by default using the option `iewopt=3` (see eq. (2.11)).

The subprocesses considered include all configurations with up to 2 light-quark pairs; they are listed in table 4, following the notation employed in the code.

For each process, the charge-conjugate ones are always understood. The above list fully covers all the possible processes with up to 3 light jets in addition to the heavy quarks. In the case of 4 or more extra jets, we do not calculate processes with 3 light-quark pairs. As in the case of associated production with a W, we expect that, within the uncertainties of the LO approximation, these can be safely neglected.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables (the cuts related to the heavy quarks are only applied in the case of b, while top quarks are always generated without cuts):

- ${p_T}^{\text{jet}}$, η^{jet}, ΔR_{jj}
- ${p_T}^{\text{Q}}$, η^{Q}, $\Delta R_{Q\overline{Q}}$
- ${p_T}^{\ell}$, η^{ℓ}, $\Delta R_{\ell j}$, $m(\ell^+\ell^-)$, for $\ell^+\ell^-$ final states
- missing E_T for $\nu\bar{\nu}$ final states.

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$u\bar{u} \to ZQ\overline{Q}$	2	$d\bar{d} \to ZQ\overline{Q}$	3	$gg \to ZQ\overline{Q}$
4	$gu \to uZQ\overline{Q}$	5	$gd \to dZQ\overline{Q}$	6	$ug \to uZQ\overline{Q}$
7	$dg \to dZQ\overline{Q}$	8	$gg \to u\bar{u}ZQ\overline{Q}$	9	$gg \to d\bar{d}ZQ\overline{Q}$
10	$u\bar{u} \to u\bar{u}ZQ\overline{Q}$	11	$d\bar{d} \to d\bar{d}ZQ\overline{Q}$	12	$uu \to uuZQ\overline{Q}$
13	$dd \to ddZQ\overline{Q}$	14	$u\bar{u} \to u'\bar{u}'ZQ\overline{Q}$	15	$d\bar{d} \to d'\bar{d}'ZQ\overline{Q}$
16	$uu' \to uu'ZQ\overline{Q}$	17	$u\bar{u}' \to u\bar{u}'ZQ\overline{Q}$	18	$dd' \to dd'ZQ\overline{Q}$
19	$d\bar{d}' \to d\bar{d}'ZQ\overline{Q}$	20	$u\bar{u} \to d\bar{d}ZQ\overline{Q}$	21	$d\bar{d} \to u\bar{u}ZQ\overline{Q}$
22	$ud \to udZQ\overline{Q}$	23	$du \to duZQ\overline{Q}$	24	$u\bar{d} \to u\bar{d}ZQ\overline{Q}$
25	$d\bar{u} \to d\bar{u}ZQ\overline{Q}$	26	$u\bar{u} \to b\bar{b}ZQ\overline{Q}$	27	$d\bar{d} \to b\bar{b}ZQ\overline{Q}$
28	$gg \to b\bar{b}ZQ\overline{Q}$	29	$gu \to uu\bar{u}ZQ\overline{Q}$	30	$ug \to uu\bar{u}ZQ\overline{Q}$
31	$gu \to uu'\bar{u}'ZQ\overline{Q}$	32	$ug \to uu'\bar{u}'ZQ\overline{Q}$	33	$gu \to ud\bar{d}ZQ\overline{Q}$
34	$ug \to ud\bar{d}ZQ\overline{Q}$	35	$gu \to ub\bar{b}ZQ\overline{Q}$	36	$ug \to ub\bar{b}ZQ\overline{Q}$
37	$gd \to dd\bar{d}ZQ\overline{Q}$	38	$dg \to dd\bar{d}ZQ\overline{Q}$	39	$gd \to dd'\bar{d}'ZQ\overline{Q}$
40	$dg \to dd'\bar{d}'ZQ\overline{Q}$	41	$gd \to du\bar{u}ZQ\overline{Q}$	42	$dg \to du\bar{u}ZQ\overline{Q}$
43	$gd \to db\bar{b}ZQ\overline{Q}$	44	$dg \to db\bar{b}ZQ\overline{Q}$		

Table 4: Subprocesses included in the $Z/\gamma^* Q\overline{Q}$+jets code. It is always understood that quarks u and d represent generic light quarks of type up or down. The Z in the table stands for a neutral $\ell^+\ell^-$ ($\nu\bar{\nu}$) lepton pair. Additional final-state gluons are not explicitly shown here but are included in the code if the requested light-jet multiplicity ($N \leq 4$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1` in the case of 2 light jets will correspond to the final state $u\bar{u} \to ZQ\overline{Q}gg$. The details can be found in the subroutine `selflav` of the file `zqqlib/zqq.f`.

	$N = 0$	$N = 1$	$N = 2$	$N = 3$	$N = 4$
LHC, (fb)	1840(5)	1085(4)	444(3)	154(2)	44(1)
FNAL, (fb)	49.3(1)	13.18(5)	2.57(2)	0.400(4)	0.0511(5)

Table 5: $\sigma(\ell^+\ell^- b\bar{b} + N \text{ jets})$ at the Tevatron and at the LHC. Parameters and cuts are given in eqs. (3.4)–(3.6).

The cut on the dilepton invariant mass allows to optimise the sampling of the DY mass spectrum if the user is interested in events off the Z peak. Additional cuts can be supplied by the user in an appropriate routine. The choice of factorization and renormalization scale is similar to what given for the $WQ\overline{Q}$ processes, with the W mass replaced by the mass of the DY pair (if the DY mass range excludes the value of m_Z), or by m_Z (when the DY mass range includes m_Z).

Some benchmark results are given in table 5, obtained for the following set of cuts and scale choice:

$$Q^2 = m_Z^2 + p_{T,Z}^2\,, \qquad 80\,\text{GeV} \leq m_{ll} \leq 100\,\text{GeV} \tag{3.4}$$

$$p_T{}^{\text{jet}} > 20\,\text{GeV}\,, \qquad |\eta_j| < 2.5\,, \quad \Delta R_{jj} > 0.7 \tag{3.5}$$

$$p_T{}^{b} > 20\,\text{GeV}\,, \qquad |\eta_b| < 2.5\,, \quad \Delta R_{b\bar{b}} > 0.7, R_{bj} > 0.7\,. \tag{3.6}$$

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$q\bar{q}' \to W$	2	$qg \to q'W$	3	$gq \to q'W$
4	$gg \to q\bar{q}'W$	5	$q\bar{q}' \to Wq''\bar{q}''$	6	$qq'' \to Wq'q''$
7	$q''q \to Wq'q''$	8	$q\bar{q} \to Wq'\bar{q}''$	9	$q\bar{q}' \to Wq\bar{q}$
10	$\bar{q}'q \to Wq\bar{q}$	11	$q\bar{q} \to Wq\bar{q}'$	12	$q\bar{q} \to Wq'\bar{q}$
13	$qq \to Wqq'$	14	$qq' \to Wqq$	15	$qq' \to Wq'q'$
16	$gg \to Wq'q''\bar{q}''$	17	$gq \to Wq'q''\bar{q}''$	18	$gq \to Wqq\bar{q}'$
19	$gq \to Wq'q\bar{q}$	20	$gq \to Wqq\bar{q}'$	21	$gq \to Wq'q\bar{q}$
22	$gg \to Wq\bar{q}'q''\bar{q}''$	23	$gg \to Wq\bar{q}q\bar{q}'$		

Table 6: Subprocesses included in the W+jets code. Additional final-state gluons are not explicitly shown here but are included in the code if the requested light-jet multiplicity ($N \leq 6$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1` in the case of 2 jet will correspond to the final state $q\bar{q}' \to Wgg$. The details can be found in the subroutine `selflav` of the file `wjetlib/wjets.f`.

3.3 W+ jets

As in the previous cases, we use the notation W as a short hand; what is actually calculated is the matrix element for a lepton+neutrino final state. All spin correlations and finite width effects are therefore accounted for. The quoted cross sections refer to a single lepton family; in the flavour assignment, the code selects by default an electron. The EW parameters are fixed by default using the option `iewopt=3` (see eq. (2.10)).

The subprocesses considered include all configurations with up to 2 light-quark pairs; they are listed in table 6, following the notation employed in the code.

For each process, the charge-conjugate ones are always understood. The above list fully covers all the possible processes with up to 3 light jets. In the case of 4 jets, we do not calculate processes with 3 light-quark pairs. Within the uncertainties of the LO approximation, these can be safely neglected [4].

As a default, the code generates kinematical configurations defined by cuts applied to the following variables:

- ${p_T}^{\rm jet}$, $\eta^{\rm jet}$, ΔR_{jj}
- ${p_T}^{\ell}$, η^{ℓ}, ${p_T}^{\nu}$, $\Delta R_{\ell j}$.

The respective threshold values can be provided by the user at run time. Additional cuts can be supplied by the user in an appropriate routine. The choice of scale follows the same conventions as for the $WQ\overline{Q}$ case.

Some benchmark results are given in table 7, obtained for the following set of cuts and scale choice:

$$Q^2 = m_W^2 + p_{T,W}^2, \tag{3.7}$$

$${p_T}^{\rm jet} > 20\,\mathrm{GeV}, \quad |\eta_j| < 2.5, \quad \Delta R_{jj} > 0.7\ . \tag{3.8}$$

For processes with up to 4 jets, we verified the numerical agreement with the results of the VECBOS code [4].

	$N=0$	$N=1$	$N=2$	$N=3$	$N=4$	$N=5$	$N=6$
LHC (pb)	18068(4)	3412(4)	1130(2)	342.9(1.4)	100.6(1.4)	27.6(4)	7.14(15)
FNAL (pb)	2087.0(6)	225.8(2)	37.3(2)	5.66(6)	0.745(4)	0.0864(15)	0.0086(2)

Table 7: $\sigma(\ell\nu + N \text{ jets})$ at the Tevatron and at the LHC. Parameters and cuts are given in eqs. (3.7)–(3.8).

`jproc`	subprocess	`jproc`	subprocess
1	$gc' \to Wc$	2	$c'g \to Wc$
3	$gg \to Wc\bar{c}'$	4	$qq'' \to Wq'q''(q'/q''=c)$
5	$q''q \to Wq'q''(q'/q''=c)$	6	$q\bar{q} \to Wq'\bar{q}''(q'/q''=c)$
7	$q\bar{q} \to Wq\bar{q}'(q/q'=c)$	8	$q\bar{q} \to Wq'\bar{q}(q/q'=c)$
9	$qq \to Wqq'(q/q'=c)$	10	$c'g \to Wcq''\bar{q}''$
11	$gc' \to Wcq''\bar{q}''$	12	$c'g \to Wc'c'\bar{c}$
13	$c'g \to Wcc'\bar{c}'$	14	$gc' \to Wc'c'\bar{c}$
15	$gc' \to Wcc'\bar{c}'$	16	$gg \to Wc\bar{c}'q''\bar{q}''$
17	$gg \to Wc'\bar{c}'c'\bar{c}$		

Table 8: Subprocesses included in the $W+c$+jets code. The symbol c' refers to either of the two Cabibbo partners of the charm quark, d or s. Additional final-state gluons are not explicitly shown here but are included in the code if the requested light-jet multiplicity ($N \leq 5$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1` in the case of 2 jet will correspond to the process $gc' \to Wcgg$. The details can be found in the subroutine `selflav` of the file `wcjetlib/wcjets.f`.

3.4 W + c+ jets

The processes included in this code are a subset of the ones treated in section 3.3. Here the final states consisting exclusively of one c (or $\bar{c}$) quark in association with a W and additional light jets are singled out. Events with *charm* quark pairs are not included, and should be generated using the `wqq` processes. All spin correlations and finite width effects in the fermionic decay of the W are accounted for. The W decay modes can be selected when running the code with `imode=2` to produce unweighted events, as discussed in the appendix B.4. The EW parameters are fixed by default using the option `iewopt=3` (see eq. (2.10)).

The subprocesses considered include all configurations with up to 2 light-quark pairs (where "light" includes the charm); they are listed in table 8, following the notation employed in the code.

For each process, the charge-conjugate ones are always understood. The above list fully covers all the possible processes with up to 2 light jets in addition to the heavy quarks. In the case of 3 extra jets, we do not calculate processes with 2 light-quark pairs. As in the case of W+jet production, we expect these to be negligible.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables:

- ${p_T}^{\text{jet}}$, η^{jet}, ΔR_{jj}
- ${p_T}^{\ell}$, η^{ℓ}, ${p_T}^{\nu}$, $\Delta R_{\ell j}$,

	$N=0$	$N=1$	$N=2$	$N=3$	$N=4$	$N=5$
LHC (pb)	418.9(8)	183.7(8)	55.6(3)	14.9(1)	3.65(3)	0.848(7)
FNAL (pb)	8.74(2)	2.39(1)	0.360(2)	0.0422(2)	0.00414(2)	0.000353(2)

Table 9: $\sigma(\ell\nu + c + N \text{ jets})$ at the Tevatron and at the LHC. Parameters and cuts are given in eqs. (3.9)–(3.10).

where the cuts on c-quarks are the same as for light jets. The respective threshold values can be provided by the user at run time. Additional cuts can be supplied by the user in an appropriate routine. The choice of scale follows the same conventions as for the $WQ\overline{Q}$ case. Some benchmark results are given in table 9, obtained for the following set of cuts and scale choice:

$$Q^2 = m_W^2 + p_{T,W}^2\,, \tag{3.9}$$

$$p_T{}^{\text{jet}} > 20\,\text{GeV}\,, \qquad |\eta_j| < 2.5\,, \qquad \Delta R_{jj} > 0.7\,. \tag{3.10}$$

3.5 $\mathbf{Z/\gamma^*}$+ jets

We use the notation Z/γ^* as a short hand; what is actually calculated is the matrix element for a charged lepton or neutrino pair final state. All spin correlations and finite width effects are therefore accounted for. When the final state $\ell^+\ell^-$ is selected, the interference between intermediate Z and γ^* is also included. The quoted cross sections refer to a single lepton family; in the flavour assignment, the code selects by default the e^+e^- pair. In the case of the final state $\nu\bar{\nu}$ the quoted cross sections include the decays to all 3 neutrino flavours, although we always label the neutrinos as ν_e. The EW parameters are fixed by default using the option `iewopt=3` (see eq. (2.11)).

All subprocesses with up to 2 light quark pairs are included. This means that the cross-sections with up to 3 final-state partons are exact. The emission of additional hard gluons can however be calculated, and the current version of the code works with up to a 6 final-state jets. The subprocesses considered are listed in table 10.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables:

- $p_T{}^{\text{jet}}$, η^{jet}, ΔR_{jj}
- $p_T{}^{\ell}$, η^{ℓ}, $\Delta R_{\ell j}$ for $\ell^+\ell^-$ final states
- missing E_T for $\nu\bar{\nu}$ final states.

Additional cuts can be supplied by the user in an appropriate routine. The choice of scale follows the same conventions as for the $ZQ\overline{Q}$ case.

Some benchmark results are given in table 11, obtained for the following set of cuts and scale choice:

$$Q^2 = m_Z^2 + p_{T,Z}^2\,, \tag{3.11}$$

$$p_T{}^{\text{jet}} > 20\,\text{GeV}\,, \qquad |\eta_j| < 2.5\,, \qquad \Delta R_{jj} > 0.7 \tag{3.12}$$

$$80\,\text{GeV} \leq m_{ll} \leq 100\,\text{GeV}\,. \tag{3.13}$$

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$u\bar{u} \to Z$	2	$d\bar{d} \to Z$	3	$gu \to uZ$
4	$gd \to dZ$	5	$ug \to uZ$	6	$dg \to dZ$
7	$gg \to u\bar{u}Z$	8	$gg \to d\bar{d}Z$	9	$u\bar{u} \to u\bar{u}Z$
10	$d\bar{d} \to d\bar{d}Z$	11	$uu \to uuZ$	12	$dd \to ddZ$
13	$u\bar{u} \to u'\bar{u}'Z$	14	$d\bar{d} \to d'\bar{d}'Z$	15	$uu' \to uu'Z$
16	$u\bar{u}' \to u\bar{u}'Z$	17	$dd' \to dd'Z$	18	$d\bar{d}' \to d\bar{d}'Z$
19	$u\bar{u} \to d\bar{d}Z$	20	$d\bar{d} \to u\bar{u}Z$	21	$ud \to udZ$
22	$du \to udZ$	23	$u\bar{d} \to u\bar{d}Z$	24	$d\bar{u} \to d\bar{u}Z$
25	$gu \to uu\bar{u}Z$	26	$ug \to uu\bar{u}Z$	27	$gu \to uu'\bar{u}'Z$
28	$ug \to uu'\bar{u}'Z$	29	$gu \to ud\bar{d}Z$	30	$ug \to ud\bar{d}Z$
31	$gd \to dd\bar{d}Z$	32	$dg \to dd\bar{d}Z$	33	$gd \to dd'\bar{d}'Z$
34	$dg \to dd'\bar{d}'Z$	35	$gd \to du\bar{u}Z$	36	$dg \to du\bar{u}Z$

Table 10: Subprocesses included in the Z/γ^*+jets code. It is always understood that quarks u and d represent generic quarks of type up or down. The Z in the table stands for a neutral $\ell^+\ell^-$ ($\nu\bar{\nu}$) lepton pair. The complex conjugate processes are also understood. Additional final-state gluons are not explicitly shown here but are included in the code if the requested jet multiplicity ($N \leq 6$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1` in the case of 2 jets will correspond to the final state $u\bar{u} \to Zgg$. The details can be found in the subroutine `selflav` of the file `zjetlib/zjets.f`.

	$N=0$	$N=1$	$N=2$	$N=3$	$N=4$	$N=5$	$N=6$
LHC (pb)	1526(1)	320.9(5)	104.6(2)	31.6(2)	9.4(2)	2.51(4)	0.65(2)
FNAL (pb)	179.4(2)	21.44(2)	3.36(1)	0.489(2)	0.0630(3)	0.00700(4)	0.000690(6)

Table 11: $\sigma(\ell^+\ell^- + N \text{ jets})$ at the Tevatron and at LHC. Parameters and cuts are given in eqs. (3.7)–(3.8).

3.6 $\mathbf{nW + mZ + lH + k}$ jets

The code computes all processes where massive EW gauge bosons and/or Higgs particles are produced on shell plus (up to 3) additional light jets. The limitations in the number of final state particles is therefore the following: $k \leq 3$ and $n + m + l + k \leq 8$. It is intended that at least one EW gauge boson or Higgs particle appear in the final state ($n+m+l > 0$). All contributions from QCD and EW processes are included. In particular, all processes of the gauge-boson-fusion type are present when the number of final state quarks is at least 2 (e.g. $qq \to qqH$). All gauge bosons are decayed to fermion pairs, taking therefore into account all spin correlations among the decay products by means of exact matrix elements. More information on gauge-invariance issues in presence of decays are discussed below, and details on the choice of the decay modes can be found in appendix B.4. The Higgs boson decays will be soon available.

In the case of $m = l = 0$ ($n = l = 0$) the code reproduces the results of the programs described in sections 3.3 and 3.5, up to the following effects: final-state finite-width corrections, are absent here since the gauge bosons are kept on shell; branching ratios depend on the selected decay mode for the Z, while they are kept to 1 in the case of the W bosons, whose final states are selected during the unweighting phase (see appendix B.4); the $\gamma^* \to \ell^+\ell^-$ contributions are only present in the code of section 3.5; EW production of jets ($W \to jj$, $Z/\gamma^* \to jj$, plus EW boson exchanges between quarks) are included here.

The EW parameters are fixed by default using the option `iewopt=3` (see eq. (2.12)). In this way we are guaranteed that all gauge boson masses are consistent with the measured values. The coupling strengths extracted from the fixed inputs, however, will differ by few percents from their best values. The user can select the option `iewopt=0` (see eq. (2.9)), where the couplings are matched to the radiatively corrected best values, at the price of working with gauge boson masses which differ by few percents from the measured masses. Both schemes are equally consistent at the LO. We verified that cross-sections obtained using the two schemes differ from each other at the level of few percent, a negligible effect compared to the large uncertainties related to the choice of factorization and renormalization scales.

The subprocesses considered are listed in table 12, where a distinction is made according to the number of final state W's. If n is odd, the whole Cabibbo structure of the matrix element is correctly taken into account. If n is even, we work in the $\cos\theta_c = 1$ approximation. The reason of this choice is that, in the latter case, different Cabibbo structures may interfere at the amplitude level. The quantity $\mathcal{F}^{\pm}$ in table 12 stands for $nW + mZ + lH$ with n odd, while $\mathcal{F}^0$ stands for $nW + mZ + lH$ when n is even.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables:

- ${p_T}^{\text{jet}}$, η^{jet}, ΔR_{jj} .

Additional cuts can be supplied by the user in an appropriate routine. Cuts on the leptons or on the Higgs decay products should be imposed in the analysis routine, using the momentum and flavour information as specified in appendix B.4.

In the code initialization phase, the user can select among 3 continuous choices for the parametrization of the factorization and renormalization scale Q: a real input parameter (`qfac`) allows to vary the overall scale of Q, $Q = \texttt{qfac} \times Q_0$, while the preferred functional form for Q_0 is selected through an integer input parameter (`iqopt`=0,1,2). In more detail:

`iqopt`	0	1	2
Q_0	$\langle M_B \rangle$	$\sum M_B$	$\sqrt{\hat{s}}$

where $\langle M_B \rangle$ and $\sum M_B$ are the average and sum of the EW boson masses (vectors as well as Higgses), respectively.

Some benchmark results are given in tables 13–18, using the scale $Q = \sum M_B$, with the sum extended over all bosons in the final state.

In order to deal, in a gauge invariant way, with the problem of the appearing unstable bosons in the intermediate states, we computed the matrix element by using zero-width propagators and by cutting away events around the mass of the unstable particle M in such a way to keep the area of a Breit-Wigner distribution. In other words, for a particle

n odd

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$q\bar{q}' \to \mathcal{F}^\pm$	2	$gq \to q'\mathcal{F}^\pm$	3	$gq \to q'\mathcal{F}^\pm$
4	$gg \to q\bar{q}'\mathcal{F}^\pm$	5	$q\bar{q}' \to \mathcal{F}^\pm q''\bar{q}''$	6	$qq'' \to \mathcal{F}^\pm q'q''$
7	$q''q \to \mathcal{F}^\pm q'q''$	8	$q''\bar{q}'' \to \mathcal{F}^\pm q\bar{q}'$	9	$q\bar{q}' \to \mathcal{F}^\pm q\bar{q}$
10	$\bar{q}'q \to \mathcal{F}^\pm q\bar{q}$	11	$q\bar{q} \to \mathcal{F}^\pm q\bar{q}'$	12	$q\bar{q} \to \mathcal{F}^\pm q'\bar{q}$
13	$qq \to \mathcal{F}^\pm qq'$	14	$qq' \to \mathcal{F}^\pm qq$	15	$qq' \to \mathcal{F}^\pm q'q'$
16	$gg \to \mathcal{F}^\pm q'q''\bar{q}''$	17	$gq \to \mathcal{F}^\pm q'q''\bar{q}''$	18	$gg \to \mathcal{F}^\pm qq\bar{q}'$
19	$gg \to \mathcal{F}^\pm q'q\bar{q}$	20	$gq \to \mathcal{F}^\pm qq\bar{q}'$	21	$gq \to \mathcal{F}^\pm q'q\bar{q}$

n even

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$q\bar{q} \to \mathcal{F}^0$	2	$gq \to q\mathcal{F}^0$	3	$gq \to q\mathcal{F}^0$
4	$gg \to q\bar{q}\mathcal{F}^0$	5	$qq \to qq\mathcal{F}^0$	6	$qq \to q'q'\mathcal{F}^0$
7	$qq' \to qq'\mathcal{F}^0$	8	$qq'' \to qq''\mathcal{F}^0$	9	$qq' \to q''q'''\mathcal{F}^0$
10	$q\bar{q} \to q\bar{q}\mathcal{F}^0$	11	$q\bar{q} \to q'\bar{q}'\mathcal{F}^0$	12	$q\bar{q} \to q''\bar{q}''\mathcal{F}^0$
13	$q\bar{q}' \to q\bar{q}'$ or $\bar{q}q'\mathcal{F}^0$	14	$q\bar{q}'' \to q\bar{q}''\mathcal{F}^0$	15	$q\bar{q}' \to q''\bar{q}'''\mathcal{F}^0$
16	$q\bar{q}'' \to q'\bar{q}'''\mathcal{F}^0$	17	$gq \to qq\bar{q}\mathcal{F}^0$	18	$gq \to qq\bar{q}\mathcal{F}^0$
19	$gq \to qq'\bar{q}'\mathcal{F}^0$	20	$gq \to qq'\bar{q}'\mathcal{F}^0$	21	$gq \to qq''\bar{q}''\mathcal{F}^0$
22	$gq \to qq''\bar{q}''\mathcal{F}^0$	23	$gq \to q'q'\bar{q}\mathcal{F}^0$	24	$gq \to q'q'\bar{q}\mathcal{F}^0$
25	$gq \to q'q''\bar{q}'''\mathcal{F}^0$	26	$gq \to q'q''\bar{q}'''\mathcal{F}^0$		

Table 12: Subprocesses included in the $nW + mZ + lH$+jets code, for the cases n=odd ($nW + mZ + lH = \mathcal{F}^\pm$) and n=even ($nW + mZ + lH = \mathcal{F}^0$). It is always understood that quarks q and q' belong to the same iso-doublet, while q and q'' belong to different iso-doublets. Additional final-state gluons are not explicitly indicated but are included in the code. The details can be found in the subroutine `selflav` of the file `vbjetslib/vbjets.f`.

of mass M and width Γ, our effective cut s_0 is determined by the equation

$$\int_{-\infty}^{M^2-s_0} ds \frac{1}{(s-M^2)^2} = \int_{-\infty}^{M^2} ds \frac{1}{(s-M^2)^2+\Gamma^2 M^2} \,. \tag{3.14}$$

that gives $s_0 = 2\Gamma M/\pi$. In practice the program discards events for which $|s - M^2| <$ `winsize` $\cdot\, \Gamma \cdot M$, with the adjustable parameter `winsize` set, by default, to the value `winsize` $= 2/\pi$. The described procedure gives sensible results when the portion of resonance cut away is of the order of a few GeV. Otherwise, holes start becoming visible in the distributions. This is the case for Z's and W's vector bosons. But problems arise for very heavy Higgses. For this reason we put a protection in the code to avoid runs when $\Gamma_H > 10\,\mathrm{GeV}$.

We checked that our algorithm reproduces, within few percents, the results obtained by using a fudge approach, defined by setting $\Gamma = 0$ in the matrix element and by multiplying the final result by the factor

$$\frac{1}{1+\left(\frac{M\Gamma}{s-M^2}\right)^2} \,.$$

	WW	WZ	ZZ
LHC (pb)	74.8(5)	28.1(2)	10.85(6)
FNAL (pb)	8.51(5)	2.45(1)	1.027(5)

Table 13: Diboson production at the Tevatron and at the LHC.

WW+jets	jj, central	jjj, central	jj, fwd	jjj,fwd
LHC, m_H=120 (pb)	18.6(2)	8.16(9)	0.256(4)	0.365(9)
LHC, m_H=200 (pb)	18.9(2)	8.3(1)	0.546(24)	0.389(9)
FNAL, m_H=120 (fb)	336(1)	49.1(2)	0.201(1)	0.0789(3)
FNAL, m_H=200 (fb)	364(2)	54.9(8)	0.415(4)	0.096(2)

Table 14: Associated production of WW and jets, at the Tevatron and at the LHC. In all cases, $E_{Tj} > 20$ GeV and $\Delta R_{jj} > 0.7$. The *central* configurations correspond to all jets with $|\eta_j| < 2.5$. The *fwd* configurations have two jets in opposite rapidity hemispheres with $2.5 < |\eta_j| < 5$, plus a central jet in the jjj case.

H+jets	jj, central	jjj, central	jj, fwd	jjj,fwd
LHC, m_H=120 (pb)	1.27(1)	0.458(6)	0.504(3)	0.089(1)
LHC, m_H=140 (pb)	0.96(1)	0.320(4)	0.458(2)	0.078(2)
LHC, m_H=200 (pb)	0.459(4)	0.132(3)	0.346(2)	0.0528(6)
FNAL, m_H=120 (fb)	121(1)	26.7(2)	0.769(2)	0.0730(5)
FNAL, m_H=140 (fb)	75.0(4)	16.0(1)	0.573(1)	0.0513(3)
FNAL, m_H=200 (fb)	21.8(2)	4.28(3)	0.246(1)	0.0193(1)

Table 15: Associated production of H and jets, at the Tevatron and at the LHC. In all cases, $E_{Tj} > 20$ GeV and $\Delta R_{jj} > 0.7$. The *central* configurations correspond to all jets with $|\eta_j| < 2.5$. The *fwd* configurations have two jets in opposite rapidity hemispheres with $2.5 < |\eta_j| < 5$, plus a central jet in the jjj case.

In the case of 3 gauge boson production, we verified the agreement with the results shown in ref. [43], up to the WZZ channel, whose rate is erroneously reported in the tables of section 5.33.[2] In the case of four boson production, a previous calculation has been documented in [44]. A comparison of the results of our code with the numbers presented in that paper shows however some discrepancies.

As a consistency check of our calculations for large multiplicities of gauge bosons, we verified that the production rates for multiple gauge bosons when the Higgs mass is above the threshold for diboson decay are well approximated by the incoherent sum of the processes mediated by an on-shell Higgs, plus those where the Higgs contribution is suppressed in the intermediate states. In the tables we show, this is for example seen in the comparison of the $WWjj$ vs Hjj or VVV vs HV rates at m_H=200 GeV.

3.7 $\mathbf{Q\overline{Q}}$+ jets

The subprocesses considered include all configurations with up to two light quark pairs.

[2] A. Ghinkulov, private communication.

JHEP07(2003)001

	WWW	WWZ	WZZ	ZZZ
LHC, m_H=120	130(1)	98(2)	31.0(4)	10.9(1)
LHC, m_H=200	305(5)	199(4)	95(1)	45.2(5)
FNAL, m_H=120	6.39(2)	5.13(4)	1.21(1)	0.519(2)
FNAL, m_H=200	18.1(2)	13.5(2)	5.47(6)	3.36(3)

Table 16: Triboson production at the Tevatron and at the LHC (fb).

	$WWWW$	$WWWZ$	$WWZZ$	$WWWWW$	$WWWWZ$
LHC, m_H=120	0.606(6)	0.72(1)	0.48(1)	$5.8(1) \cdot 10^{-3}$	$10.8(3) \cdot 10^{-3}$

Table 17: 4- and 5-boson production at the LHC (fb).

WH	m_H=120	m_H=140	m_H=200	ZH	m_H=120	m_H=140	m_H=200
LHC (pb)	1.364(8)	0.833(4)	0.251(2)		0.727(4)	0.449(2)	0.137(1)
FNAL (fb)	122.4(6)	70.0(2)	16.55(4)		75.0(3)	44.2(2)	11.24(3)

Table 18: H+boson production at the Tevatron and LHC.

They are listed in table 19, following the notations employed in the code. The list covers all possible processes involving up to 4 light-parton jets with 2 light quark pairs. Subprocesses involving 3 light-quark pairs, which would only appear in the case of 4 jets in addition to the heavy quarks, are not included, as they are expected to contribute a negligible rate.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables (the cuts related to the heavy quarks are only applied in the case of b, while top quarks are always generated without cuts):

- ${p_T}^{\rm jet}$, $\eta^{\rm jet}$, $\Delta R_{jj}, \Delta R_{jb}$
- ${p_T}^b$, η^b, $\Delta R_{b\bar{b}}$.

The respective threshold values can be provided by the user at run time. Additional cuts can be supplied by the user in an appropriate routine. In the case $Q = t$ the decay of $t\bar{t}$ pairs in six fermions (in the on-shell approximation) is enforced. With this option the decay of the top quarks takes into account all spin correlations among the decay products by means of exact matrix elements. More information on the selection of decay products can be found in appendix B.4.

In the code initialization phase, the user can select between 2 choices for the parameterization of the factorization and renormalization scale Q. A real input parameter (`qfac`) allows to vary the overall scale of Q, $Q =$ `qfac` $\times Q_0$, while the preferred functional form for Q_0 is selected through an integer input parameter (`iqopt`=0,1). In more detail:

`iqopt`	0	1
$Q_0[tt]^2$	m_t^2	$m_t^2 + \langle p_T^2 \rangle$
$Q_0[bb]^2$	$\hat{s}$	$\langle p_T^2 \rangle$

where $\langle p_T \rangle$ is the average p_T of light partons, if $Q = t$, while is the average transverse momentum of all final state jets in the case of $Q = b$.

`jproc`	subprocess	`jproc`	subprocess	`jproc`	subprocess
1	$gg \to Q\overline{Q}$	2	$q\bar{q} \to Q\overline{Q}$	3	$gq \to Q\overline{Q}q$
4	$qg \to Q\overline{Q}q$	5	$gg \to Q\overline{Q}q\bar{q}$	6	$q\bar{q} \to Q\overline{Q}q\bar{q}$
7	$q\bar{q} \to Q\overline{Q}q'\bar{q}'$	8	$qq' \to Q\overline{Q}qq'$	9	$q\bar{q}' \to Q\overline{Q}q\bar{q}'$
10	$qq \to Q\overline{Q}qq$	11	$gg \to Q\overline{Q}qq'\bar{q}'$	12	$qg \to Q\overline{Q}qq'\bar{q}'$
13	$gq \to Q\overline{Q}qq\bar{q}$	14	$qg \to Q\overline{Q}qq\bar{q}$	15	$gg \to Q\overline{Q}q\bar{q}q\bar{q}$
16	$gg \to Q\overline{Q}q\bar{q}q'\bar{q}'$				

Table 19: Subprocesses included in the $Q\overline{Q}$+jets code. Additional final-state gluons are not explicitly shown here but are included in the code if the requested light-jet multiplicity ($N \leq 6$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1`, in the case of 2 extra jets, will correspond to the final state $gg \to Q\overline{Q}gg$. The details can be found in the subroutine `selflav` of the file `2Qlib/2Q.f`. For each process, the charge-conjugates ones are always understood.

$Q\overline{Q} + N$ jets	$N = 0$	$N = 1$	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N = 6$
$Q = t$, LHC (pb)	530.0(8)	462.6(6)	255(1)	111.5(5)	42.4(4)	14.07(16)	4.36(8)
$Q = t$, FNAL (fb)	6,364(8)	1,592(3)	282(1)	40.6(3)	4.83(4)	0.483(6)	0.0419(9)
$Q = b$, LHC (nb)	1,533(4)	422(1)	130.2(6)	30.9(4)	7.5(4)	1.53(6)	0.337(9)
$Q = b$, FNAL (pb)	72,1(1)	12,15(2)	2,51(1)	365(7)	47.3(9)	5.6(2)	0.58(2)

Table 20: $\sigma(Q\overline{Q} + N \text{ jets})$ at the Tevatron and at the LHC, with parameters and cuts given in eqs. (3.15)–(3.18).

Some benchmark results are given in table 20, obtained for the following set of cuts and scale choice:

$$t\bar{t}: \quad m_t = 175\,\text{GeV}\,, \qquad Q^2 = m_t^2 \tag{3.15}$$

$$b\bar{b}: \quad m_b = 4.75\,\text{GeV}\,, \qquad Q^2 = \frac{(p_{T,b}^2 + p_{T,\bar{b}}^2 + \sum p_{T\,j}^2)}{(2+N)} \tag{3.16}$$

$$p_T{}^{\text{jet}} > 20\,\text{GeV}\,, \qquad |\eta_j| < 2.5\,, \qquad \Delta R_{jj} > 0.7 \tag{3.17}$$

$$p_T{}^{b} > 20\,\text{GeV}\,, \qquad |\eta_b| < 2.5\,, \qquad \Delta R_{b\bar{b}} > 0.7\,, \Delta R_{bj} > 0.7\,. \tag{3.18}$$

In the case of 0 and 1 jet, we find agreement with the results obtained using the $\mathcal{O}(\alpha_s^3)$ code of ref. [45].

3.8 $\mathbf{Q\overline{Q}Q'\overline{Q}'}$+ jets

The list of available processes is given in table 21. This covers all possible processes up to 2 light jets and at most 1 light quark pair. The cases with one additional heavy quark pair (e.g. $t\bar{t}b\bar{b}b\bar{b}$) are also included. Subprocesses with two light-quark pairs give a negligible contribution to the case of 2 extra jets, and are not calculated. For the parameterization of the factorization and renormalization scale the user has two choices, `iqopt`=0,1, as described in the following table:

JHEP07(2003)001

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$gg \to Q\overline{Q}Q'\overline{Q}'$	2	$q\bar{q} \to Q\overline{Q}Q'\overline{Q}'$	3	$gq \to Q\overline{Q}Q'\overline{Q}'q$
4	$gg \to Q\overline{Q}Q'\overline{Q}'q$	5	$gg \to Q\overline{Q}Q'\overline{Q}'q\bar{q}$	6	$gg \to Q\overline{Q}Q'\overline{Q}'b\bar{b}$

Table 21: Subprocesses included in the $Q\overline{Q}Q'\overline{Q}'+$ jets code. Additional final-state gluons are not explicitly shown here but are included in the code if the requested light-jet multiplicity ($N \leq 4$) exceeds the number of indicated final-state partons. For example, the subprocess `jproc=1`, in the case of 2 extra jets, will correspond to the final state $gg \to Q\overline{Q}Q'\overline{Q}'gg$. The details can be found in the subroutine `selflav` of the file `4Qlib/4Q.f`. For each process, the charge-conjugates ones are always understood.

`iqopt`	0	1
$Q_0[bb]^2$	$\hat{s}$	$\langle p_T^2 \rangle$
$Q_0[bt]^2$	$\hat{s}$	m_t^2
$Q_0[tt]^2$	$\hat{s}$	m_t^2

where $\langle p_T^2 \rangle$ is the average p_T^2 of light partons and b's.

Again, as a default, the code generates kinematical configurations defined by cuts applied to the following variables (the cuts related to the heavy quarks are only applied in the case of b, while top quarks are always generated without cuts):

- ${p_T}^{\rm jet}$, $\eta^{\rm jet}$, $\Delta R_{jj}, \Delta R_{jb}$
- ${p_T}^b$, η^b, $\Delta R_{b\bar{b}}$.

The respective threshold values can be provided by the user at run time. Additional cuts can be supplied by the user in an appropriate routine.

Some benchmark results are given in table 22, obtained for the following set of cuts and scale choice: m_t=175 GeV, m_b=4.75 GeV,

$$t\bar{t}t\bar{t} \text{ and } t\bar{t}b\bar{b}: \quad Q^2 = m_t^2 \tag{3.19}$$

$$b\bar{b}b\bar{b}: \quad Q^2 = \frac{(p_{T,b}^2 + p_{T,\bar{b}}^2 + \sum p_{Tj}^2)}{(2+N)} \tag{3.20}$$

$$ {p_T}^{\rm jet} > 20\,\text{GeV}\,, \qquad |\eta_j| < 2.5\,, \qquad \Delta R_{jj} > 0.7 \tag{3.21}$$

$$ {p_T}^b > 20\,\text{GeV}\,, \qquad |\eta_b| < 2.5\,, \qquad \Delta R_{b\bar{b}} > 0.7\,, \qquad \Delta R_{bj} > 0.7\,. \tag{3.22}$$

3.9 $\mathbf{Q\overline{Q}H}$+ jets

The list of processes is given in table 23. The Higgs is produced only via Yukawa couplings to the heavy quarks, no other EW process is included. All cases with up to 2 light-quark pairs are included, covering in full the possible final states with up to 3 jets in addition to the QQH system. The code will otherwise deal with up to 4 extra jets.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables:

- ${p_T}^{\rm jet}$, $\eta^{\rm jet}$, $\Delta R_{jj}, \Delta R_{jb}$

$Q\overline{Q}Q'\overline{Q}' + N$ jets	$N=0$	$N=1$	$N=2$	$N=3$	$N=4$
$t\bar{t}t\bar{t}$, LHC (fb)	12.73(8)	17.4(2)	13.5(1)	7.55(6)	3.48(5)
$t\bar{t}b\bar{b}$, LHC (pb)	1.35(1)	1.47(2)	0.94(2)	0.457(8)	0.189(4)
$t\bar{t}b\bar{b}$, FNAL (fb)	3.44(3)	0.95(1)	0.154(1)	0.0187(2)	0.00187(5)
$b\bar{b}b\bar{b}$, LHC (pb)	477(2)	259(5)	95(1)	28.6(6)	25.0(3)
$b\bar{b}b\bar{b}$, FNAL (pb)	6.64(5)	2.25(3)	0.470(5)	0.076(1)	0.0025(5)

Table 22: $\sigma(Q\overline{Q}Q'\overline{Q}' + N \text{ jets})$ at the Tevatron and at the LHC, with parameters and cuts given in eqs. (3.19)–(3.22).

`jproc`	subprocess	`jproc`	subprocess	`jproc`	subprocess
1	$gg \to Q\overline{Q}H$	2	$q\bar{q} \to Q\overline{Q}H$	3	$gq \to Q\overline{Q}qH$
4	$qg \to Q\overline{Q}qH$	5	$gg \to Q\overline{Q}q\bar{q}H$	6	$gg \to Q\overline{Q}b\bar{b}H$
7	$qq \to Q\overline{Q}qqH$	8	$qq' \to Q\overline{Q}qq'H$	9	$q\bar{q} \to Q\overline{Q}q\bar{q}H$
10	$q\bar{q}' \to Q\overline{Q}q\bar{q}'H$	11	$q\bar{q} \to Q\overline{Q}q'\bar{q}'H$	12	$q\bar{q} \to Q\overline{Q}b\bar{b}H$
13	$gq \to Q\overline{Q}q\bar{q}qH$	14	$qg \to Q\overline{Q}q\bar{q}qH$	15	$gq \to Q\overline{Q}q'\bar{q}'qH$
16	$qg \to Q\overline{Q}q'\bar{q}'qH$	17	$gq \to Q\overline{Q}b\bar{b}qH$	18	$qg \to Q\overline{Q}b\bar{b}qH$

Table 23: Subprocesses included in the $Q\overline{Q}H$ code. The details can be found in the subroutine `selflav` of the file `QQhlib/QQh.f`. For each process, the charge-conjugate subprocesses are always understood.

- ${p_T}^b$, η^b, $\Delta R_{b\bar{b}}$.

The respective threshold values can be provided by the user at run time. Additional cuts can be supplied by the user in an appropriate routine. In the case $Q = t$ the decay of $t\bar{t}$ pairs in six fermions (in the on-shell approximation). With this option the decay of the top quarks takes into account all spin correlations among the decay products by means of exact matrix elements. More information on how to use the selection of decay products can be found in appendix B.4. The Higgs boson decay will be soon available.

For the parameterization of the factorization and renormalization scale the user has three choices, `iqopt`=0,1,2, as described in the following table:

`iqopt`	0	1	
$Q_0[bb]^2$	$\hat{s}$	$m_H^2 + \langle p_T^2 \rangle$	$\langle p_T^2 \rangle$
$Q_0[tt]^2$	$\hat{s}$	$(2m_t + m_H)^2$	$\langle p_T^2 \rangle$

where $\langle p_T^2 \rangle$ is the average ${p_T}^2$ of light and heavy partons. Some benchmark results are given in table 24, obtained for the following set of cuts and scale choice:

$$t\bar{t}: \; Q^2 = (2m_t + m_H)^2 \tag{3.23}$$

$$b\bar{b}: \; Q^2 = m_H^2 + \frac{(p_{T,b}^2 + p_{T,\bar{b}}^2)}{2}\,. \tag{3.24}$$

As a default, the Yukawa coupling of the Higgs to the bottom quarks is evaluated at the b-quark pole mass. Since the rate is directly proportional to y_b^2, the result corresponding

$Q\overline{Q}H$	$m_H = 120\,\mathrm{GeV}$	$m_H = 150\,\mathrm{GeV}$	$m_H = 200\,\mathrm{GeV}$
$Q = t$, LHC (fb)	401(2)	212(1)	89.1(4)
$Q = t$, FNAL (fb)	4.22(2)	2.00(1)	0.637(4)
$Q = b$, LHC (fb)	599(3)	279(3)	99(2)
$Q = b$, FNAL (fb)	3.73(3)	1.20(1)	0.240(2)

Table 24: $\sigma(Q\overline{Q}H)$, at the Tevatron and at the LHC, with parameters given in eqs. (3.23)–(3.24). No cuts applied.

to the choice $y_b \propto m_b(m_H)$ can be obtained via a trivial rescaling. The numbers we obtain in the case of 0 extra jets agree with what found in the literature [46, 47], after possibly correcting for the difference between pole and running b-mass.

3.10 N jets

The subprocesses considered include all configurations with up to two light quark pairs. They are listed in table 25, following the notations employed in the code. The list covers all possible processes involving up to 6 light-parton jets with 2 light quark pairs. Subprocesses involving 3 light-quark pairs, which only appear for 4 or more jets, are not included, but are expected to contribute a negligible rate (they are fully included in the `NJETS` code by Berends et al [5]). In both initial and final states we only assume as quark types u, d, s and c. For the generation of events with heavier quarks (b and t), we suggest using the `2Q` element of the package.

As a default, the code generates kinematical configurations defined by cuts applied to the following variables:

- ${p_T}^{\mathrm{jet}}$, η^{jet}, ΔR_{jj}.

The respective threshold values can be provided by the user at run time. Additional cuts can be supplied by the user in an appropriate routine. In the code initialization phase, the user can select between 2 choices for the parameterization of the factorization and renormalization scale Q. A real input parameter (`qfac`) allows to vary the overall scale of Q, $Q =$ `qfac` $\times Q_0$, while the preferred functional form for Q_0 is selected through an integer input parameter (`iqopt`=0,1). In more detail:

`iqopt`	0	1
Q_0^2	$\hat{s}$	$\langle p_T \rangle$

where $\langle p_T \rangle$ is the average p_T of the final state jets.

Some benchmark results are given in table 26, obtained for the following set of cuts and scale choice:

$$Q^2 = \langle p_T \rangle \tag{3.25}$$

$${p_T}^{\mathrm{jet}} > 20\,\mathrm{GeV}\,, \qquad |\eta_j| < 2.5\,, \qquad \Delta R_{jj} > 0.7\,. \tag{3.26}$$

jproc	subprocess	jproc	subprocess	jproc	subprocess
1	$gg \to gg$	2	$q\bar{q} \to gg$	3	$gq \to gq$
4	$qg \to qg$	5	$gg \to q\bar{q}$	6	$qq \to qq$
7	$qq' \to qq'$	8	$q\bar{q}' \to q\bar{q}'$	9	$q\bar{q} \to q\bar{q}$

Table 25: Subprocesses included in the Njets code. $N-2$ additional final-state gluons are not explicitly shown here but are included in the code, with N up to 6. For example, the subprocess `jproc=4`, in the case of $N=4$ corresponds to the final state $qg \to qggg$. The details can be found in the subroutine `selflav` of the file `Njetslib/Njets.f`. For each process, the charge-conjugates ones are always understood.

N jets	$N=2$	$N=3$	$N=4$	$N=5$	$N=6$
LHC (nb)	$375(1)\cdot 10^3$	$24.5(2)\cdot 10^3$	4,174(8)	709(1)	126.3(4)
FNAL (nb)	23,706(60)	857(5)	90.9(1)	8.66(1)	$826(1)\cdot 10^{-3}$

Table 26: $\sigma(N$ jets) at the Tevatron and at the LHC, with parameters and cuts given in eqs. (3.25)–(3.26).

4. Conclusions

We presented in this paper a new MC tool for the generation of complex, high-multiplicity hard final states in hadronic collisions. To the best of our knowledge, a large fraction of the processes we discussed have never been calculated before in the literature to the level of jet multiplicities considered here, due to the complexity of the matrix elements involved. In addition to the evaluation of the matrix elements, and the possibility of performing complete parton-level simulations, the code we developed offers the possibility to carry out the shower evolution and hadronization of the partonic final states. In the current version we implemented the Les Houches format for the event representation, and developed the relative interface with HERWIG. In the future other hard processes (for example including emission of real, hard photons) will be added to the list of available reactions and Higgs decay to two or four fermions will be included.

Our code will allow complete and accurate studies of the SM backgrounds to a large fraction of the most interesting new physics phenomena accessible at the Tevatron, at the LHC, and at future high-energy hadron colliders.

Acknowledgments

The work of M.L.M. and F.P. is supported in part by the EU Fourth Framework Programme "Training and Mobility of Researchers", Network "Quantum Chromodynamics and the Deep Structure of Elementary Particles", contract FMRX–CT98–0194 (DG 12 – MIHT). M.M. and R.P. acknowledge the financial support of the European Union under contract HPRN-CT-2000-00149. A.D.P. is supported by a M. Curie fellowship, contract HPMF-CT-2001-01178.

We thank P. Richardson and B. Webber for invaluable help in the development of the Les Houches compliant HERWIG interface for HERWIG, and A. Messina for his contribution

JHEP07(2003)001

to the PYTHIA interface. FP thanks G. Montagna and O. Nicrosini for their collaboration during the early stage of the development of the $Q\overline{Q}Q'\overline{Q}'$ processes, and the Pavia Gruppo IV of INFN for access to the local computing resources.

A. The contents of the code package

The code is written in Fortran77, with the part relative to the matrix element evaluation available as well in Fortran 90 (see appendix C). The code package, contained in the compressed file `alpgen.tar.gz`, can be obtained from the URL
`http://home.cern.ch/mlm/alpgen`.

Unpacking the zipped tarred file `alpgen.tar.gz` with the command:

```
> tar -zxvf alpgen.tar.gz
```

will create the following directory structure:

```
2Qlib/     Njetswork/  herlib/     wcjetwork/  zjetlib/
2Qwork/    QQhlib/     pylib/      wjetlib/    zjetwork/
4Qlib/     QQhwork/    vbjetslib/  wjetwork/   zqqlib/
4Qwork/    VF90/       vbjetswork/ wqqlib/     zqqwork/
Njetslib/  alplib/     wcjetlib/   wqqwork/
```

The directories labeled `'lib'` contain the source codes for the respective processes. The user is not supposed to touch them. The code elements which the user will need to access and possibly modify to run his own analyses are contained in the directories labeled `'work'`. More in detail:

- The directory `alplib/` contains the parts of code which are generic to the evaluation of matrix elements using the ALPHA algorithm. The user should treat this directory as a black box. When new processes will be added in the future, this part of the code should not change. More in detail:
  - `alplib/alpgen.f`: contains the general structure of the code, preparing the input for the matrix element calculation, the bookkeeping of the cross-section determination, the event generation, etc.
  - `alplib/alpgen.inc`: include file, with the necessary common blocks.
  - `alplib/Aint.f, Asu3.f, Acp.f`: the set of programmes necessary for the calculations of the matrix element, done by the ALPHA algorithm.
  - `alplib/alppdf.f`: contains a collection of structure function parameterizations; some of them require at run time input tables, which are provided as part of the package, and stored in the subdirectory `alplib/pdfdat/`. The command file `alplib/pdfdat/hvqpdf` contains the necessary logical links to all PDF data tables. As a default, we already provide a logical link to this file in all `/*work` directories. It is sufficient to issue the `pdflnk` command within the desired

working subdirectory to create the necessary logical links, and allow the use of all available PDFs.

  - `alplib/alputi.f`: This program unit contains a histogramming package which allows to generate `topdrawer` [49] files with the required distributions. Examples of the use of this package are provided in the default user files `*work/*usr.f`. Users who prefer other histogramming packages, such as HBOOK, do not need to link to this file.

- The directories `*lib/` (`*=wqq`, `zqq`, `wcjet`, `wjet`, `zjet`, `vbjets`, `2Q`, `4Q`, `QQh`, `Njets`) contain the parts of the code specific to the generation of $WQ\overline{Q}+$ jets, $ZQ\overline{Q}+$ jets, $W+$ jets, $W+c+$ jets, $Z+$ jets, $nW+mZ+lH+$ jets, $Q\overline{Q}+$ jets, $Q\overline{Q}Q'\overline{Q}'+$ jets $Q\overline{Q}H+$ jets and N jets events. The respective `include` files with the necessary process-dependent common blocks are included in these directories. The user should treat these directories as black boxes.

- The directories `*work/` (`*=wqq`, `zqq`, `wcjet`, `wjet`, `zjet`, `vbjets`, `2Q`, `4Q`, `QQh`, `Njets`) contain the parts of the code which the user is supposed to interact with, in order to implement his own analysis cuts, etc. They contain the files `*usr.f`, where sample analysis routines are provided. These files host the routines in which the user can select generation mode, generation parameters (e.g. beam energy, PDF sets, heavy quark mass, etc.) as well as generation cuts (minimum p_T thresholds, etc.). Here the user initialises the histograms, writes the analysis routine, and prints out the required program output. This is the only part of the code in which the user is supposed to operate, editing the analyses files, and producing and running the executable (see next section). The versions provided as a default contain already complete running examples, with the respective command files (`input`) containing sets of default settings (see next section for more details on running the code).

- The directory `herlib/` contains the parts of code relevant for the shower evolution using HERWIG. In addition to the HERWIG source and include files for version 6.4, this directory includes the file `atoher.f`, which is the interface between the parton-level matrix elements and HERWIG, and the file `hwuser.f`, which includes the main driver for the running of HERWIG, and the part of the code where the user can input the analyses routines. More in detail:

  - `hwuser.f`: user initialization of the analysis. Includes the standard HERWIG initialization, histogram intialization, analysis routines, etc. The calls to new routines `hwigup` and `hwupro` create the interface with the generated hard events.

  - `atoher.f`: this file contains all routines necessary to read in the unweighted events produced by the hard matrix element generator. The routine `hwigup` downloads the initialization parameters of the hard process (process type, number of partons, beam energy and beam type, etc.), and allows the main HERWIG initialization. The routine `hwupro` is called for each event: it reads the event kinematics, flavour and colour information from the file of unweighted events, and translates the event data to allow the HERWIG processing of the shower. This file should be treated by the user as a black box.

– `herwig64.f`: the `herwig64` source code.

– `HERWIG64.INC, herwig6400.inc`: HERWIG common blocks

– `pdfdummy.f`: dummy PDF routine, required by HERWIG unless the CERN library PDF sets are used. As a default, the current version runs with the default HERWIG PDF set, regardless of the PDF set which was used to generate the hard process. We verified that this does not affect the features of the showered final state. Nevertheless we plan in the next version to enforce the consistency between PDF set used in the hard generation and in the shower evolution.

- A similar directory and file structure is provided for PYTHIA.

B. Running the code

To compile the code for the $WQ\overline{Q}$ process,[3] change directory to:

```
> cd wqqwork
```

A `Makefile` is provided for compilation. Issue the comand

```
> make wqqgen
```

and the executable `wqqgen` will be prepared. It can be run interactively, inputting from the keyboard the run parameters requested by the code, or using the default command file `input`, issuing the command

```
> wqqgen < input
```

Editing the file `input` allows to change the initialization defaults (e.g. the number of jets, the heavy quark masses, the PDF sets, etc.). Notice that altering some of the inputs in the `input` file may influence the sequence of parameters requested from the code; for example, depending on whether light jets are requested or not in a given process, the generation cuts for these jets will or will not be required in input. The `input` file should therefore be edited in a consistent manner.

The first input parameter requested is the running mode `imode`. The three available running modes are discussed in detail in the following subsections.

The codes `vbjets`, `2Q` and `QQh` include the option `idecay`, to allow for the decay of the generated on shell vector bosons and top quarks, taking into account spin correlations between fermion decay products by means of decay matrix elements. More details on how to run the codes are discussed in the dedicated subsection.

[3] Analogous procedures allow compilation and run of other processes.

JHEP07(2003)001

B.1 `imode=0`

The simplest option is `imode=0`, where events are generated according to the selected cuts, a total cross section is evaluated, and the user can use the routine `evtana` to analyse the event and fill histograms with the desired distributions. To facilitate the job of the user, we provide a (redundant) array of kinematical variables relative to the event. The array is initialised in the routine `usrfll` contained in the file `wqqlib/wqq.f`, and is stored in the common block `usrevt` contained in the include file `wqq.inc`. Examples of variables provided include:

- `pin(4,2)`: momenta of the incoming partons
- `pout(4,8)`: momenta of the outgoing particles (maximum 8 outgoing partons)
- `pjet(4,8)`: momenta of the final-state partons (i.e. quarks and gluons)
- `ptj(8), etaj(8)`: transverse momentum and pseudorapidity of the final-state partons
- `pbott(4) (pbbar(4)), ptb (ptbb)`: momentum and transverse momentum of the heavy (anti)quark
- `plep(4) (ptlep), pnu(4) (ptmiss)`: (transverse) momentum of the charged lepton and neutrino
- `drjj(8,8)`: ΔR separation in $\eta - \phi$ space among the final-state partons
- `drbj(8) (drbbj(8))`: ΔR separation in $\eta - \phi$ space between the heavy (anti)quark, and the final-state partons
- etc…

Similar sets of variables are provided for the other processes. As an output the user will find the following files: (`'file'` is the label assigned by the user at run-start time):

- `file.stat`: the header of this file contains information on the run: value of the input parameters (EW and strong couplings, beam types and energies, PDF set), hard process selected and generation cuts. Furthermore, this file reports the results of each individual integration cycle, with total cross sections, as well as individual contributions from the allowed subprocesses. It gives the maximum weights of the various iterations, and the corresponding unweighting efficiencies, and the value of the cross-section accumulated over the various iterations, weighted by the respective statistical errors.
- `file.top`: includes the topdrawer plots of the distributions, if requested. The default normalization of the histograms is in pb/bin.
- `cnfg.dat`: file required by ALPHA, generated at run time; it is not needed for the analysis of the output, and will be recreated anew any time the code runs, so the user should not bother about it, and it can be safely deleted at any time.

- `file.mon`: produced/updated after each 100K events; it contains information on the status of the run, dumped every 100K events. It is useful to monitor the progress of the run. In addition to this monitoring tool, the user can choose to perform other tests during the run, in order to save partial information, or monitor the evolution of the plotted distributions. Every 100K events the program calls the routine `monitor`, contained in the user file `wqqwork/wqqusr.f`, where the user can select which operations to perform. As a default, the provided routine prints out each 1M events the topdrawer file with the distributions being histogrammed. In case of crash, the results relative to the statistics accumulated up to that point are therefore retrievable.

- `file.grid*`: The phase-space is discretised and parameterised by a multi-dimensional grid. During the phase-space integration, a record is kept of the rate accumulated within each bin of each integration variable. At the end of an integration cycle ("iteration"), the total bin-by-bin rates are used to improve the grid sampling efficiency. This is achieved by assigning sampling probabilities proportional to the bin integrals (we ensure however that 20% of the sampling is uniformly distributed among all bins, to avoid artificial biases introduced by runs with limited statistics). A subsequent iteration can then benefit from a better sampling. The state of the grid at the end of each iteration is saved in the file `file.grid1`. Since the first few iterations give rise to distributions which are likely to be biased by large statistical fluctuations, we separate a phase of grid warm-up from a phase in which events will be generated and distributions calculated. The user should then specify in the `input` file the number of warm-up iterations, the number of events to be calculated for each iteration, and then the number of events that will be used for the final event generation and for the analysis. At the end of the event generation, the grid will be saved to the file `file.grid2`. We keep this separate from `file.grid1` to allow the user to choose whether to start a new generation cycle using the grid status at the end of the previous warm-up phase, or at the end of the previous generation phase. These choices are made by the user in the `input` file, selecting the variable `igrid` to be 0 (to reset the grids and start a new grid optimization), 1 (to start the new run using the grid obtained during the previous warm-up phase) or 2 (to start the new run using the grid optimised at the end of the latest event generation). Both grid files are saved in `file.grid*-old` (*=1,2) at the beginning of each run, to allow recovery of the grid information in case of run crashes or mistakes.

B.2 `imode=1`

Running the code with `imode=1` offers the same functionality than `imode=0`, but will in addition write the weighted events to a file. To limit the size of the file, only events which passed a pre-unweighting are saved. The pre-unweighting is based on a maximum weight w_{tmp} which is equal to 1% of the actual maximum weight at the moment of the generation of the event: $w_{tmp} = w_{\max}/100$. An event with weight w passing the pre-unweighting is then assigned a weight $w' = w_{tmp}$ if $w < w_{tmp}$, or $w' = w$ if $w > w_{tmp}$. The weight w' is then saved to a file, together with the random number seed which initiated the generation

JHEP07(2003)001

of this event, and with the value of x_1 (useful to check the sanity of the file when it will be read again for the unweighting). Some statistical information on the run, including the total number of generated events, the integral, and the overall maximum weight, are saved as well in a separate file. The file with weighted events is to be used for a later unweighting. One can easily verify that the pre-unweighting procedure does not introduce any bias in the final unweighting. The random number seed will then be sufficient to regenerate the full kinematical, flavour and colour information on the event. The size of each event is 57 bytes. Make sure you have enough disk space to write out the number of events you require. In addition to the files listed above for `imode=0`, as an output the user will find the following files:

- `file.par`: includes run parameters (e.g. beam energies and types, generation cuts, etc), phase-space grids, cross-section and maximum-weight information;
- `file.wgt`: for each event we store the two seeds of the random number generation, the event weight, and the value of x_1 for the event (as a sanity benchmark after the kinematics has been reconstructed from the random seeds)

As a default, the bookkeeping of the weight distribution is kept in the routine `*usr.f`, and the relative data are printed in the topdrawer file `file.top`. The study of the weight distribution can guide the user to a more efficient choice of maximum weight before starting the event unweighting.

B.3 `imode=2`

After a run with `imode=1`, a run with `imode=2` will perform the unweighting of the already generated events, and will prepare the input file for the processing of the events with HERWIG or PYTHIA. The code reads first the phase-space grids used for the weighted-event generation and the maximum weight from `file.par`. The user has the possibility to edit `file.par` and replace the maximum weight with a different value, if he convinces himself that a more efficient unweighting can be obtained, without biasing the sample, by selecting a smaller maximum weight.[4] We are working on techniques to perform this optimised unweigthing in an automatic way. The code will then scan the file `file.wgt`, containing the weighted events. A comparison of the event weights with the maximum weight is made, and the unweighting is performed. The kinematics of each unweighted event is reconstructed from the relative random number seed. The colour flow for the event is then calculated, and the full event information is written to a new file. This file will be the starting point for the generation of the full shower, to be performed using HERWIG or PYTHIA. As an output the user will find the following files:

- `file_unw.stat`: includes cross sections, max weight, etc;
- `file_unw.top`: while no new events are generated, the analysis routines used when running in `imode=1` are applied to the unweighted events, and the relative distribu-

[4] The user must however avoid using a maximum weight smaller than 1% of the true maximum weight, because of the threshold used in the pre-unweighting phase.

tions are evaluated. In this way the user can compare distributions before and after unweighting;

- `file.unw`: list of unweighted events, including event kinematics, flavour and colour structure, and event weight.

B.4 Decay of top quarks and vector bosons with spin correlations

The on-shell top quarks generated by `2Q` and `QQh` undergo a fully exclusive decay in three fermions weighted with exact matrix element. When running in `imode=0,1` the information on top decay product momenta is stored in the matrix `idec(4,3,2)`, available in the routines `usrfll` and `usrcut`. The meaning of the entries is as follows: `idec(1:4,i,j)` is the four momentum (p_x, p_y, p_z, E) of the i-th decay product of the j-th particle (`jtl` and `jtbl` are the labels for top and antitop quark respectively); `i=1` is the label for the b ($\bar{b}$) quark; `i=2,3` are the labels for the fermion and antifermion coming from the W decay respectively. When running in `imode=2` the user is required (interactively) to select one top decay mode among seven options:

$$
\begin{aligned}
\mathtt{1} &= e\nu_e b\bar{b} + 2\text{jets}\,,\\
\mathtt{2} &= \mu\nu_\mu b\bar{b} + 2\text{jets}\,,\\
\mathtt{3} &= \tau\nu_\tau b\bar{b} + 2\text{jets}\,,\\
\mathtt{4} &= l\nu_l b\bar{b} + 2\text{jets}\,, \qquad (l = e, \mu, \tau)\\
\mathtt{5} &= l\nu_l l'\nu_{l'} b\bar{b}\,, \qquad (l = e, \mu, \tau)\\
\mathtt{6} &= b\bar{b} + 4\ \text{jets}\,,\\
\mathtt{7} &= \text{fully inclusive}\,.
\end{aligned}
$$

Non-zero masses of the W-decay products are introduced by rescaling their momenta, and keeping invariant the W 4-momentum.

The on-shell W and Z gauge bosons generated in `vbjets` undergo a fully exclusive decay. While the decay products of the W can be changed when running in `imode=2` (see below), given the universal electroweak couplings of the W to fermions, the Z decay options must be specified at the very beginning. When running in `imode=0,1` the information on the W, Z decay product momenta is stored in the matrix `idec(4,4,maxpar-2)`, available in the routines `usrfll` and `usrcut`. The meaning of the entries is as follows: `idec(1:4,i,j)` is the four momentum (p_x, p_y, p_z, E) of the i-th decay product of the j-th particle; `i=1,2` are the labels for fermion and antifermion respectively. The ordering in `j` correspond to the ordering in which Z and W's are generated. The flavour of the Z decay products is stored in the variable `zfl(maxpar-2)` according to PDG conventions. For every Z boson in the final state the user have to select its decay mode in the input file by entering an integer string with the decay modes of the individual Z's according to the following table

$$
\begin{aligned}
\mathtt{1} &= \nu\bar{\nu}\,, \qquad \text{(summed over all flavours)}\,,\\
\mathtt{2} &= l^+l^-\,, \qquad \text{(summed over all flavours)}\,,\\
\mathtt{3} &= q\bar{q}\,, \qquad \text{(summed over all flavours)}\,,
\end{aligned}
$$

$$
\begin{aligned}
\texttt{4} &= b\bar{b}\,,\\
\texttt{5} &= \text{fully inclusive}\,.
\end{aligned}
$$

Concerning the decay modes of the W's, they have to be specified when running in `imode=2` as follows (the code asks for this automatically and interactively):

$$
\begin{aligned}
\texttt{1} &= e\bar{\nu}_e\,,\\
\texttt{2} &= \mu\bar{\nu}_\mu\,,\\
\texttt{3} &= \tau\bar{\nu}_\tau\,,\\
\texttt{4} &= l\bar{\nu}_l \qquad (l = e, \mu, \tau)\,,\\
\texttt{4} &= q\bar{q}'\,,\\
\texttt{5} &= \text{fully inclusive}\,.
\end{aligned}
$$

The same final state options for W bosons are available also for `wjets` and `wqq` with `imode=2`. Finite fermionic masses of the decay products are introduced by rescaling the momenta and preserving the vector boson 4-momentum.

B.5 Running HERWIG or PYTHIA on the unweighted events

A HERWIG executable can be obtained starting from the default driver `herlib/hwuser.f`. To compile and link, issue the command:

```
> make hwuser
```

from the `herlib` directory. The resulting executable, `hwuser`, should then be run in the directory containing the unweighted event file. The default code simply generates the shower evolution for the given events, and prints to the screen the particle content of the first few events (number of printed events set by the variable `maxpr`). Analysis code should be provided by the user, by filling the standard HERWIG routines `hwabeg, hwanal, hwaend`. A log file `file-her.log` documenting the inputs and outputs of the run is produced. An analogous interface for running PYTHIA is available.

C. Portability and Fortran 90

The code was tested on several platforms, including Linux based PC's, Digital Alpha Unix, HP series 9000/700, Sun work stations and MAC-OSX with a `g77` compiler. The user should however check that the compilation options provided by default in the first few lines of the `Makefile` files, including the choice of Fortran compiler, are consistent with what he has available. We shall be happy to receive comments related to the portability of the code, and will update the code to improve its usability.

Toghether with the old Fortran77 (`F77`) version of the ALPHA code [33, 25], we provide a new Fortran90 (`F90`) version. The evaluation of the ALPHA matrix elements with the `F90` version is a factor of five to twenty times faster than the `F77` version, depending on the selected process (the more complex the process, the better the improvement). When the overhead of the rest of the code (phase-space, parton densities, etc) is added in, the overall performance of the code improves by a factor of two to five (we stress that only the ALPHA part of the code is available in `F90`; users unfamiliar with `F90` should not be discouraged from using this version, since this component is a black box, and its use is compatible with the `F77` part of the code which the user has access to. Furthermore, the `F90` executables will run using the same `input` files as the `F77` versions of the code, and produce the same results, to machine precision). The improved performance is essentially due to a major rewriting of the ALPHA code avoiding redundancies present in the previous version.

To link the F90 version of ALPHA it is sufficient to input the choice of `F90` compiler in the `Makefile`, and issue the comand `make wqqgen90`, which will produce the executable `wqqgen90`.

For user who do not have access to a `F90` compiler, we provide one suitable for running on PC's with Linux operating systems. To set it up, proceed as follows:

- go to the `ALPGEN` home directory;
- issue the command `make ft90V`. This will unpack the file `ft90V.tar.gz` and install the *Vast/Verydian* `F90` compiler into the directory `F90V`. This software was distributed freeware for personal use only by Pacific-Sierra Research. Before use, you are therefore supposed to agree with the license term contained in the directory `F90V/`. In the same directory the user can find some documentation on the compiler, including the list of supported platforms.
- Move to the desired work directory (e.g. `wqqwork`);
- issue the command `make wqqgen90V` which will produce the executable `wqqgen90V`.

References

[1] For a review of multi-parton processes in QCD, see M.L. Mangano and S.J. Parke, *Multiparton amplitudes in gauge theories*, *Phys. Rept.* **200** (1991) 301.

[2] F. Gianotti et al., *Physics potential and experimental challenges of the LHC luminosity upgrade*, `hep-ph/0204087`.

[3] I. Hinchliffe, *The Papageno partonic Monte Carlo program*, LBL-34372, submitted to *Workshop on Physics at Current Accelerators and the Supercollider*, Argonne, IL, 2-5 Jun 1993.

[4] F.A. Berends, H. Kuijf, B. Tausk and W.T. Giele, *On the production of a W and jets at hadron colliders*, *Nucl. Phys.* **B 357** (1991) 32.

[5] F.A. Berends, W.T. Giele and H. Kuijf, *On six jet production at hadron colliders*, *Phys. Lett.* **B 232** (1989) 266.

[6] P.D. Draggiotis, R.H.P. Kleiss and C.G. Papadopoulos, *Multi-jet production in hadron collisions*, *Eur. Phys. J.* **C 24** (2002) 447 [`hep-ph/0202201`].

[7] P.D. Draggiotis and R. Kleiss, *Parton counting: physical and computational complexity of multi-jet production at hadron colliders*, *Eur. Phys. J.* **C 17** (2000) 437 [`hep-ph/0006133`].

JHEP07(2003)001

[8] T. Stelzer and W.F. Long, *Automatic generation of tree level helicity amplitudes*, *Comput. Phys. Commun.* **81** (1994) 357 [hep-ph/9401258].

[9] A. Pukhov et al., *Comphep: a package for evaluation of Feynman diagrams and integration over multi-particle phase space. User's manual for version 33*, hep-ph/9908288.

[10] MINAMI-TATEYA GROUP collaboration, T. Ishikawa et al., *GRACE manual: automatic generation of tree amplitudes in Standard Models: version 1.0*, KEK-92-19.

[11] F. Krauss, R. Kuhn and G. Soff, *AMEGIC++ 1.0: a matrix element generator in C++*, *J. High Energy Phys.* **02** (2002) 044 [hep-ph/0109036].

[12] G. Marchesini and B.R. Webber, *Monte carlo simulation of general hard processes with coherent QCD radiation*, *Nucl. Phys.* **B 310** (1988) 461;
G. Marchesini et al., *Herwig: a Monte Carlo event generator for simulating hadron emission reactions with interfering gluons. Version 5.1 - april 1991*, *Comput. Phys. Commun.* **67** (1992) 465;
G. Corcella et al., *HERWIG 6: an event generator for hadron emission reactions with interfering gluons (including supersymmetric processes)*, *J. High Energy Phys.* **01** (2001) 010 [hep-ph/0011363].

[13] T. Sjostrand, *High-energy physics event generation with PYTHIA 5.7 and JETSET 7.4*, *Comput. Phys. Commun.* **82** (1994) 74;
T. Sjostrand et al., *High-energy-physics event generation with PYTHIA 6.1*, *Comput. Phys. Commun.* **135** (2001) 238 [hep-ph/0010017].

[14] F.E. Paige, S.D. Proto pescu, H. Baer and X. Tata, *ISAJET 7.40: a Monte Carlo event generator for pp, $\bar{p}p$ and e^+e^- reactions*, hep-ph/9810440.

[15] M.H. Seymour, *Matrix element corrections to parton shower algorithms*, *Comput. Phys. Commun.* **90** (1995) 95 [hep-ph/9410414];
G. Corcella and M.H. Seymour, *Initial state radiation in simulations of vector boson production at hadron colliders*, *Nucl. Phys.* **B 565** (2000) 227 [hep-ph/9908388];
G. Miu and T. Sjostrand, *W production in an improved parton shower approach*, *Phys. Lett.* **B 449** (1999) 313 [hep-ph/9812455].

[16] G. Corcella and M.H. Seymour, *Matrix element corrections to parton shower simulations of heavy quark decay*, *Phys. Lett.* **B 442** (1998) 417 [hep-ph/9809451];
G. Corcella, M.L. Mangano and M.H. Seymour, *Jet activity in $t\bar{t}$ events and top mass reconstruction at hadron colliders*, *J. High Energy Phys.* **07** (2000) 004 [hep-ph/0004179].

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

[18] Y. Kurihara et al., *QCD event generators with next-to-leading order matrix-elements and parton showers*, *Nucl. Phys.* **B 654** (2003) 301 [hep-ph/0212216].

[19] S. Frixione and B.R. Webber, *Matching NLO QCD computations and parton shower simulations*, *J. High Energy Phys.* **06** (2002) 029 [hep-ph/0204244].

[20] S. Frixione, P. Nason and B. Webber, work in progress.

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

[22] S. Catani, F. Krauss, R. Kuhn and B.R. Webber, *QCD matrix elements + parton showers*, *J. High Energy Phys.* **11** (2001) 063 [hep-ph/0109231].

[23] F. Krauss, *Matrix elements and parton showers in hadronic interactions*, *J. High Energy Phys.* **08** (2002) 015 [hep-ph/0205283].

[24] See talks by F. Krauss, S. Mrenna and P. Richardson at the Fermilab Workshop on MC tunings, April 29-30 2003, http://cepa.fnal.gov/CPD/MCTuning/.

[25] F. Caravaglios, M.L. Mangano, M. Moretti and R. Pittau, *A new approach to multi-jet calculations in hadron collisions*, *Nucl. Phys.* **B 539** (1999) 215 [hep-ph/9807570].

[26] M.L. Mangano, M. Moretti and R. Pittau, *Multijet matrix elements and shower evolution in hadronic collisions: $Wb\bar{b}+n$ jets as a case study*, *Nucl. Phys.* **B 632** (2002) 343 [hep-ph/0108069].

[27] A.S. Belyaev et al., *Comphep-pythia interface: integrated package for the collision events generation based on exact matrix elements*, hep-ph/0101232.

[28] K. Sato et al., *Integration of GRACE and PYTHIA*, hep-ph/0104237.

[29] B.P. Kersevan and E. Richter-Was, *The Monte Carlo event generator ACERMC version 1.0 with interfaces to PYTHIA 6.2 and HERWIG 6.3*, hep-ph/0201302.

[30] S. Tsuno et al., *Gr ppa_4b: a four bottom quark production event generator for PP/$p\bar{p}$ collisions*, *Comput. Phys. Commun.* **151** (2003) 216 [hep-ph/0204222].

[31] F. Maltoni and T. Stelzer, *Madevent: automatic event generation with madgraph*, *J. High Energy Phys.* **02** (2003) 027 [hep-ph/0208156].

[32] E. Boos et al., *Generic user process interface for event generators*, hep-ph/0109068; W. Giele et al., *The QCD/SM working group: summary report*, hep-ph/0204316.

[33] F. Caravaglios and M. Moretti, *An algorithm to compute born scattering amplitudes without Feynman graphs*, *Phys. Lett.* **B 358** (1995) 332 [hep-ph/9507237].

[34] P. Draggiotis, R.H.P. Kleiss and C.G. Papadopoulos, *On the computation of multigluon amplitudes*, *Phys. Lett.* **B 439** (1998) 157 [hep-ph/9807207].

[35] H.L. Lai et al., *Improved parton distributions from global analysis of recent deep inelastic scattering and inclusive jet data*, *Phys. Rev.* **D 55** (1997) 1280 [hep-ph/9606399].

[36] CTEQ collaboration, H.L. Lai et al., *Global QCD analysis of parton structure of the nucleon: CTEQ5 parton distributions*, *Eur. Phys. J.* **C 12** (2000) 375 [hep-ph/9903282].

[37] J. Pumplin et al., *New generation of parton distributions with uncertainties from global QCD analysis*, *J. High Energy Phys.* **07** (2002) 012 [hep-ph/0201195].

[38] A.D. Martin, R.G. Roberts, W.J. Stirling and R.S. Thorne, *Parton distributions and the LHC: W and Z production*, *Eur. Phys. J.* **C 14** (2000) 133 [hep-ph/9907231].

[39] A.D. Martin, R.G. Roberts, W.J. Stirling and R.S. Thorne, *MRST2001: partons and α_s from precise deep inelastic scattering and Tevatron jet data*, *Eur. Phys. J.* **C 23** (2002) 73 [hep-ph/0110215].

[40] A.D. Martin, R.G. Roberts, W.J. Stirling and R.S. Thorne, *NNLO global parton analysis*, *Phys. Lett.* **B 531** (2002) 216 [hep-ph/0201127].

[41] Z. Kunszt, *Associated production of heavy Higgs boson with top quarks*, *Nucl. Phys.* **B 247** (1984) 339.

[42] M.L. Mangano, *Production of W plus heavy quark pairs in hadronic collisions*, *Nucl. Phys.* **B 405** (1993) 536;
R.K. Ellis and S. Veseli, *Strong radiative corrections to $Wb\bar{b}$ production in $p\bar{p}$ collisions*, *Phys. Rev.* **D 60** (1999) 011501 [hep-ph/9810489].

[43] S. Haywood et al., *Electroweak physics*, hep-ph/0003275.

[44] V.D. Barger, T. Han and H. Pi, *Four weak boson production at e^+e^- and pp supercolliders*, *Phys. Rev.* **D 41** (1990) 824.

[45] M.L. Mangano, P. Nason and G. Ridolfi, *Heavy quark correlations in hadron collisions at next-to-leading order*, *Nucl. Phys.* **B 373** (1992) 295.

[46] Higgs Working Group collaboration, M. Carena et al., *Report of the Tevatron Higgs working group*, hep-ph/0010338.

[47] M. Spira, Higgs production code available from the URL:
http://home.cern.ch/mspira/hqq/. The code is based on matrix elements from [41, 48].

[48] J.F. Gunion, *Phys. Lett.* **B 253** (1991) 269;
W.J. Marciano and F.E. Paige, *Associated production of Higgs bosons with $t\bar{t}$ pairs*, *Phys. Rev. Lett.* **66** (1991) 2433;
D.A. Dicus and S. Willenbrock, *Higgs boson production from heavy quark fusion*, *Phys. Rev.* **D 39** (1989) 751;
M. Spira, *QCD effects in Higgs physics*, *Fortschr. Phys.* **46** (1998) 203.

[49] Computation Research Group, R.B. Chaffee, *Topdrawer*, SLAC, CGTM No. 178.

JHEP07(2003)001