Preprint typeset in JHEP style - PAPER VERSION

November 15, 2020
POWHEG-BOX-V2

The Z/γ^* EW NLO & QCD production in the POWHEG-BOX-V2 user manual: svn v3616

Mauro Chiesa
INFN, Sezione di Pavia, Via A. Bassi 6, 27100 Pavia, Italy
E-mail: Mauro.Chiesa@physik.uni-wuerzburg.de

Homero Martinez
Dipartimento di Fisica, Università di Pavia and INFN, Sezione di Pavia, Via A. Bassi 6, 27100 Pavia, Italy
E-mail: Homero.Martinez@pv.infn.it

Guido Montagna
Dipartimento di Fisica, Università di Pavia and INFN, Sezione di Pavia, Via A. Bassi 6, 27100 Pavia, Italy
E-mail: Guido.Montagna@pv.infn.it

Paolo Nason
INFN, Sezione di Milano-Bicocca, Piazza della Scienza 3, 20126 Milan, Italy
E-mail: Paolo.Nason@mib.infn.it

Oreste Nicrosini
INFN, Sezione di Pavia, Via A. Bassi 6, 27100 Pavia, Italy
E-mail: Oreste.Nicrosini@pv.infn.it

Fulvio Piccinini
INFN, Sezione di Pavia, Via A. Bassi 6, 27100 Pavia, Italy
E-mail: Fulvio.Piccinini@pv.infn.it

Alessandro Vicini
Dipartimento di Fisica, Università di Milano and INFN, Sezione di Milano, Via Celoria 16, I-20133 Milano, Italy
E-mail: Alessandro.Vicini@mi.infn.it

Abstract: This note documents the use of the package POWHEG-BOX-V2 for Z/γ^* production processes including QCD and ElectroWeak NLO corrections. Results can be easily interfaced to shower Monte Carlo programs, in such a way that both NLO and shower accuracy are maintained.

Keywords: POWHEG, Shower Monte Carlo, NLO, Electroweak.

Contents

1. Introduction

The `POWHEG BOX` program is a framework for implementing NLO calculations in Shower Monte Carlo programs according to the `POWHEG` method. An explanation of the method and a discussion of how the code is organized can be found in Refs. [1, 2, 3]. The code is distributed according to the "MCNET GUIDELINES for Event Generator Authors and Users" and can be found at the web page

`http://powhegbox.mib.infn.it`.

This program is an implementation of the Drell-Yan NLO cross sections $pp \to Z/\gamma^* \to \ell^+\ell^-$ including QCD and ElectroWeak (EW) radiative corrections. A detailed description of the implementation can be found in Ref. [4]. Major improvements w.r.t. svn version 3616 in Ref. [6]. In order to run the `POWHEG BOX` program, we recommend the reader to start from the `POWHEG BOX` user manual, which contains all the information and settings that are common between all subprocesses. In this note we focus on the settings and parameters specific to the Z/γ^* implementation.

2. Generation of events

Build the executable

```
$ cd POWHEG-BOX-V2/Z_ew-BMNNPV
$ make pwhg_main
```
[1]

Then do (for example)

[1]Version 3812 or newer is working with POWHEG-BOX-V2 version 3812 or newer.

```
$ cd runtest-lhc-8TeV
$ ../pwhg_main
```

At the end of the run, the file `pwgevents.lhe` will contain 1000 events for $Z/\gamma^* \to e^+e^-$ in the Les Houches format (`.lhe` file).

We provide four executables, called "interfaces", which process the events in the file `pwgevents.lhe` and give them as inputs to parton shower programs, performing the required vetoes and setting the required flags in a manner consistent with the physical accuracy of the input events. `PYTHIA` is used to perform the QCD shower. The QED shower can be done using `PYTHIA` or the independent program `PHOTOS` [7]. The four interfaces are:

- `main-PYTHIA81-lhef` : In this interface, `PYTHIA8` (versions 8.1xx) is used to perform the QCD shower, while the QED shower can be done using `PHOTOS++` (C++ version of `PHOTOS`), or `PYTHIA8`. It is designed to work with versions of `PYTHIA8` up to 8.186. It has been tested with `PYTHIA` version 8.185 and `PHOTOS++` version 3.56. The output of the shower can be analyzed looking at the histograms in the output `.top` file. Optionally, the events after the shower can be saved in another `.lhe` file (see section 3).

  Even if you do not use `PHOTOS++`, it has to be linked. The minimal procedure is to use the version 3.56 included in the `Z_ew-BMNNPV` package, doing:

  ```
  $ cd PHOTOS
  $./configure --without-hepmc
  $ make
  ```

  The flag `PHOTOSCC_LOCATION` must be set in the `Makefile` to the path of installation of `PHOTOS++`. This path can be the folder `Z_ew-BMNNPV/PHOTOS` if the user wants to use the included version of `PHOTOS++`, otherwise it should be set to the folder of an external installation.

  `PYTHIA8` has to be downloaded and compiled by the user. The script `pythia8-config` should configure automatically the path to the `PYTHIA8` installation in the `Makefile`, if this is not the case, the user must set manually the `PYTHIA8LOCATION` flag.

  Once `PHOTOS++` and `PYTHIA8` are compiled and the flags are set properly in the `Makefile`, the interface can be compiled, doing:

  ```
  $ cd POWHEG-BOX-V2/Z_ew-BMNNPV
  $ make main-PYTHIA81-lhef
  ```

  Before running, the path to `PHOTOS++` libraries needs to be added to the list of dynamically linked libraries, and a variable pointing to the path of `PYTHIA8` particle data (`.xml`) files) needs to be set. In order to do this, the script `setlibrarypaths.sh` must be edited to point to the correct paths, and then executed in the current shell, doing:

```
$ source setlibrarypaths.sh
```

To run the interface, do:

```
$ cd runtest-lhc-14TeV-wp
$ ../main-PYTHIA81-lhef
```

- `main-PYTHIA82-lhef` : Similar interface to `main-PYTHIA81-lhef`, but designed to work with newer versions of `PYTHIA` (versions 8.2xx). It has been tested with version 8.223. The compilation / linking procedure is similar to the one described in the previous item.

- `main-PYTHIA6-lhef` : Interface to `PYTHIA6` and `PHOTOS` (both codes written in Fortran). The source codes of `PYTHIA6` and `PHOTOS` are included in the `POWHEG BOX` and in the `Z_ew-BMNNPV` folder, respectively, so no flag or linking to external libraries need to be done. To compile and run, simply do:

```
$ cd POWHEG-BOX-V2/Z_ew-BMNNPV
$ make main-PYTHIA6-lhef
$ cd runtest-lhc-14TeV
$ ../main-PYTHIA6-lhef
```

- `main-PHOTOS-lhef` : Interface to `PHOTOS++`, it processes the `POWHEG` generated events, calls the QED final state shower implemented by `PHOTOS++` and generates a new event file in LHE format, that can be then interfaced to a QCD shower program, where QED radiation must be switched off to avoid double counting. The compilation and linking requires the `PHOTOS` related steps described in item (1).

The customization of the histograms and cuts provided in the output `.top` file (for interfaces `main-PYTHIA81-lhef`, `main-PYTHIA82-lhef` and `main-PYTHIA6-lhef`), can be done modifying the code in `pwhg_analysis.f` (`analysis` subroutine). In addition, C++ based analysis can be done accessing the events from the files `pythia81F77.cc`, `pythia82F77.cc` (function `pythia_next_`) or in the file `photosCCF.cc` (function `photos_process_`), for the interfaces `main-PYTHIA81-lhef`, `main-PYTHIA82-lhef` and `main-PHOTOS-lhef`, respectively.

3. Process specific input parameters

All the parameters and flags are set in the input card file `powheg.input`. The mandatory parameters are those needed to select the final state leptonic species coming from the vector-boson:

```
vdecaymode 11 !  code for selected Z decay
              !  (11(-11):  electronic; 13(-13):  muonic; 15(-15):  tauonic)
```

The decay $Z \to \nu\bar{\nu}$ is not handled in the present version.

In addition to the mandatory parameters, the POWHEG BOX input allows for an easy setting of EW and run parameters, by explicitly adding the relevant lines to the input card. If one of the following entries is not present in the input card the reported default value is assumed. In any case, these parameters are printed in the output of the program, so their values can be easily tracked down.

```
Wmass   80.398         !  W mass in GeV
Wwidth  2.141          !  W width in GeV
Zmass   91.1876        !  Z mass in GeV
Zwidth  2.4952         !  Z width in GeV
alphaem 0.00729735254  !  em coupling alpha(0)
gmu     1.16637d-5     !  Fermi constant in GeV^-2
Hmass   120.           !  Higgs mass in GeV
Tmass   172.9          !  Top mass in GeV
Bmass     4.6          !  B quark mass in GeV
Cmass     1.2          !  C quark mass in GeV
Smass    0.15          !  S quark mass in GeV
Umass    0.06983       !  U quark mass in GeV
Dmass    0.06984       !  D quark mass in GeV
Elmass   0.005109989   !  Electron mass in GeV
Mumass   0.105658369   !  Mu mass in GeV
Taumass  1.77699       !  Tau mass in GeV
```

The following parameters limits from below the virtuality of the Z boson:

```
mass_low 20         !  Z virtuality > mass_low in GeV
```

If absent, it is set to 30 GeV. In order to avoid edge effects, the lower limit mass_low should be more inclusive w.r.t. cuts applied at the analysis level. Notice that, if photons are generated, the Z virtuality is not necessarily the mass of the dilepton.

```
mass_high 1000         !  Z virtuality < mass_high in GeV
```

If absent, it is set to $\sqrt{s}$. Upper limit on the Z virtuality: as explained before, this option can be used only if EW corrections are turned off (no_ew option, see below).

```
runningscale 0     !  choice for ren and fac scales in Bbar integration
                               0:  fixed scale M_Z
```

```
                                   1:  running scale ℓ+ℓ−(γ) inv mass
                                       γ included with QED FSR
```

With running scale, a minimum cutoff of 5 GeV is imposed on $m(\ell^+\ell^-)$.

```
scheme 1!  choice for EW NLO scheme calculation
                                   0:  Alpha(0)
                                   1:  Alpha(M_Z)
                                   2:  G_μ
                                   3:  Alpha(q^2)
```

The CKM mixing matrix is assumed diagonal in the EW NLO corrections.

The EW radiative corrections can be calculated according to four different schemes, with two different realizations:
the $\alpha(0)$ scheme, `scheme 0`, where the input parameters are $\alpha(0)$, M_W and M_Z;

the $\alpha(M_Z^2)$ scheme, `scheme 1`, where the input parameters are $\alpha(M_Z^2)$, M_W and M_Z (with this scheme the value of the parameter `alphaem_z` should be specified);

the G_μ scheme, `scheme 2`, where the input parameters are G_μ, M_W and M_Z;

a modified version of the $\alpha(M_Z^2)$ scheme, `scheme 3`, where α is computed at the scale $q^2 = (p_q + p_{\bar{q}})^2$ where p_q and $p_{\bar{q}}$ are the momenta of the incoming partons in the underlying Born kinematics (`kn_pborn`).

An additional option is to replace M_W with $\sin^2\vartheta^\ell_{eff}$ as input parameter, by means of `use-s2effin numerical value` of $\sin^2\vartheta^\ell_{eff}$.

Note that in scheme 1, 2, and 3 the the electroweak corrections are proportional to $\alpha^2\alpha_0$ with α_0 =`alphaem`.

The EW corrections can be switched off by setting
`no_ew 1 !  default 0`
and the strong corrections can be switched off by setting
`no_strong 1 !  default 0`
This last option is just to check EW corrections at the NLO level (i.e., the Les Houches events do not have much meaning).

`emalpharunning 1 !  default 0`
If `emalpharunning` is set to one, a running value is employed for α in the POWHEG Sudakov exponent for QED radiation. The running effectively accounts for the radiation of undetected e^+e^- and $\mu^+\mu^-$ pairs. If this option is used, no pair-radiation should be

allowed at the Parton Shower level.

`complexmasses 1 !  default 0`
If `complexmasses` is set to one, the calculation is performed in the complex mass scheme.

`QED-only 1 !  default 0`
If `QED-only` is set to one, only the QED part of the $O(\alpha)$ corrections in computed (the purely weak part of the virtual EW corrections is not included in the calculation).

`weak-only 1 !  default 0`
If `weak-only` is set to one, only the purely weak part of the $O(\alpha)$ corrections is computed. When the flags `weak-only`, `no_strong` and `LOevents` are set to 1 at the same time, POWHEG integrates the Born matrix element plus the purely weak NLO corrections.

`ew_ho 1 !  default 0`
If `ew_ho` is set to one, the leading part of the higher order (H.O.) purely weak corrections are computed and added to the NLO EW results. The corrections are written in terms of the quantities $\Delta\alpha$ and $\Delta\rho$. We implemented a modified version of the formulas in Ref. [5] that takes into account the running of α up to the virtuality of the Z boson (set the flag `constantscale` to 1-default 0- to use $\Delta\alpha(M_Z^2)$ instead of $\Delta\alpha(q^2)$). When using the G_μ scheme (`scheme` 2), `constantscale` 1 is automatically set for consistency at two loops.

`ew_ho_only 1 !  default 0`
If `ew_ho_only` is set to one, only the H.O. weak corrections are computed (i.e. they are not added to the NLO virtual EW ones). If the flags `ew_ho_only`, `ew_ho`, `no_strong` and `LOevents` are set to 1 at the same time, POWHEG integrates the Born matrix element plus the purely weak H.O. corrections. These setting should be used only for fixed order studies.

`kt2minqed 0.000001 !  default 0.000001`
Lower cutoff on the relative transverse momentum of the radiated photons at the LesHouches level. To ensure a proper matching with the parton shower, the parameters `SI_kt2minqed` and `kt2minqed` should have the same value.

`da_had_from_fit 1 !  default 0`
If `da_had_from_fit` is set to 0 the hadronic running of α is computed perturbatively

from the quark masses. It is possible to compute the hadronic running of alpha from the $e^+e^- \to$ hadrons data by setting the flag `da_had_from_fit` to 1 in combination with the flag `fit` (default 0). In particular, if `fit` is set to 1 the parametrization of Ref. [8] is used, while if `fit` is set to 2 the parametrization of Ref. [9, 10] is employed (if `fit` is set to 0, the running is still computed perturbatively: this option was introduced for testing purposes only).

3.1 Flags used by the shower interfaces

The flags starting with `SI_` configure the behavior of the shower interfaces. The following flags are used by the interfaces `main-PYTHIA81-lhef` and `main-PYTHIA82-lhef`:

`SI_inputfile`: Configure the input file for the shower interfaces (default: `./pwgevents.lhe`).

`SI_maxshowerevents`: Number of events to read from the input file (default: all events).

`SI_pythiamatching`, explained below.

`SI_pytune`: Set the `PYTHIA` tune used in the QCD shower.

`SI_dopythiaqed`: Turn ON / OFF `PYTHIA` QED shower (default OFF).

`SI_use_photos`: Turn ON / OFF the final state QED shower by `PHOTOS` (default OFF). If it is ON, it automatically sets off the final state QED radiation from `PYTHIA8`.

`SI_kt2minqed`: Set value of photos low energy cut off (default is 10^{-6}).

`SI_usepy8veto`, explained below.

`SI_nohad`: Allows to switch OFF the hadronization (the hadronization is ON by default).

`SI_savelhe`: Turn ON / OFF the production of an output LHE file (default OFF).

`SI_savehistos`: Turn ON / OFF the production of histograms in `.top` file (default OFF).

`SI_no_tworadevents`, explained below.

According to the `POWHEG` method, the radiation by the shower has to be generated from a starting scale given by the hardest p_{T} tried at the matrix element level. Traditionally, this scale is written in the variable `scalup` in the tt LHE event file. However, for the DY process, when both QCD and EW NLO corrections are present (Ref. [4]), it is necessary to keep track in the `POWHEG` events of two scales, one for initial state radiation (`scalup-isr`) and one for final state radiation (`scalup-fsr`). These two scales are used as starting points for the IS and FS showers, respectively. `PYTHIA8` and `PHOTOS++` do not use `scalup-fsr` for the generation of QED final state radiation from the Z. Hence, in order to avoid double counting of QED radiation, a veto algorithm is necessary. This veto is activated unless the flag `no_ew` is activated in `powheg.input`.

A general issue is the matching between the NLO calculation and the (QCD and QED) higher order corrections given by the parton shower: due to the different definitions of $p_\perp$ in POWHEG and PYTHIA8, some double counting or dead zone can arise. By default (flag `SI_pythiamatching` equal to 2), the interface generates all QCD/QED shower emissions up to the kinematical limit and then veto emissions harder than the POWHEG scales (`scalup-isr` and `scalup-fsr`), computed according to the POWHEG $p_\perp$ definition. The user can optionally choose an alternative scheme, where the shower starting scales are fixed and no veto's are performed. This choice can be activated by setting the flag `SI_pythiamatching` equal to 1.

When FS QED radiation is implement by PHOTOS++, a Fortran-based function is used to veto QED emissions and perform the correct matching (implemented in the file `main-PYTHIA-8.f`). If FS QED radiation is done by PYTHIA8, the matching can be done using Fortran-based algorithms coded in the files `main-PYTHIA-8.f` and `scalupveto.f` (flag `SI_usepy8veto` equal to 0), or alternatively, use the matching algorithms provided by PYTHIA8 (flag `SI_usepy8veto` equal to 1). Those algorithms use the `UserHooks` functions, and their behavior depend on the value of the flag `SI_pythiamatching`.

In the case the interface is used to read events where the two scales `scalup-isr` and `scalup-fsr` are not present, and `scalup` is to be used as starting scale for the showers, the flag `SI_no_tworadevents` should be set to 1.

The decay of hadronic resonances which can proceed radiatively has been suppressed. In order to let the resonances decay, the user should open the file `pythia81F77.cc` (or `pythia82F77.cc`) and comment the relevant lines in the function `pythia_init`.

The interface `main-PYTHIA6-lhef` can be customized using the following flags: `SI_inputfile`, `SI_maxshowerevents`, `SI_pytune`, `SI_use_photos`, `SI_kt2minqed`, `SI_savehistos`, `SI_no_tworadevents`, read also from the `powheg.input` file. The other flags have no effect. Notice that in this case, the QED radiation from PYTHIA is ON by default.

The interface `main-PHOTOS-lhef` uses only the relevant flags, namely: `SI_inputfile`, `SI_maxshowerevents`, `SI_kt2minqed`.

For further customization of the settings used by the shower interfaces, beyond the flags available in `powheg.input`, the user can modify the following source code files:

- Interface `main-PYTHIA81-lhef`: Settings in file `pythia81F77.cc`.
- Interface `main-PYTHIA82-lhef`: Settings in file `pythia82F77.cc`.
- Interface `main-PYTHIA6-lhef`: Settings in file `setup-PYTHIA6-lhef.f`
- Interface `main-PHOTOS-lhef`: Settings in file `photosCCF.cc`.

References

[1] P. Nason, "A new method for combining NLO QCD with shower Monte Carlo algorithms," JHEP **0411** (2004) 040 [arXiv:hep-ph/0409146].

[2] S. Frixione, P. Nason and C. Oleari, "Matching NLO QCD computations with Parton Shower simulations: the POWHEG method," JHEP **0711** (2007) 070 [arXiv:0709.2092 [hep-ph]].

[3] S. Alioli, P. Nason, C. Oleari and E. Re, "A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX," [arXiv:1002.2581 [hep-ph]].

[4] L. Barzè, G. Montagna, P. Nason, O. Nicrosini, F. Piccinini and A. Vicini, Eur. Phys. J. C **73** (2013) 2474 [arXiv:1302.4606[hep-ph]].

[5] S. Dittmaier and M. Huber, JHEP **1001**, 060 (2010) doi:10.1007/JHEP01(2010)060 [arXiv:0911.2329 [hep-ph]].

[6] C.M. Carloni Calame, M. Chiesa, H. Martinez, G. Montagna, O. Nicrosini, F. Piccinini, and A. Vicini, [arXiv:1612.02841[hep-ph]].

[7] P. Golonka and Z. Was, Eur. Phys. J. C **45** (2006) 97 [hep-ph/0506026].

[8] S. Eidelman and F. Jegerlehner, Z. Phys. C **67**, 585 (1995) doi:10.1007/BF01553984 [hep-ph/9502298].

[9] K. Hagiwara, R. Liao, A. D. Martin, D. Nomura and T. Teubner, J. Phys. G **38**, 085003 (2011) doi:10.1088/0954-3899/38/8/085003 [arXiv:1105.3149 [hep-ph]].

[10] A. Keshavarzi, D. Nomura and T. Teubner, Phys. Rev. D **97**, no. 11, 114025 (2018) doi:10.1103/PhysRevD.97.114025 [arXiv:1802.02995 [hep-ph]].