DYNNLO version 1.5

This is a note about the DYNNLO program. DYNNLO is a parton level Monte Carlo program that computes the cross section for vector-boson production in pp and $p\bar{p}$ collisions. The calculation is performed up to NNLO in QCD perturbation theory. The program includes $\gamma - Z$ interference, finite-width effects, the leptonic decay of the vector boson and the corresponding spin correlations. The user is allowed to apply arbitrary (though infrared safe) cuts on the final state and to plot the corresponding distributions in the form of bin histograms. If you use this program, please quote Ref. [1].

1 Introduction

The DYNNLO program is based on an extension of the subtraction formalism to NNLO, as described in Ref. [1].

The calculation is organized in two parts. In the first part (virtual), the contribution of the regularized virtual corrections (up to two-loop order) is computed. In the second part (real), the cross section for the production of the vector boson in association with (at least) one jet is first evaluated up to NLO (i.e. up to $\mathcal{O}(\alpha_S^2)$). This step of the calculation can be performed with any available version of the subtraction method. Here we use the dipole formalism [3], as implemented in the MCFM Monte Carlo program [4]. Since the V + jet cross section is divergent when the transverse momentum q_T of the vector boson becomes small, a suitable counterterm must be subtracted to make the result finite as $q_T \to 0$. The program uses the counterterm introduced in the second paper of Ref. [1], and thus it completes the evaluation of the real part. Finally, the two contributions (virtual and real) are combined to obtain the full cross section.

In the present version of the code, the evaluation of the subtraction counterterm is made substantially faster than before.

The program can be downloaded from `http://theory.fi.infn.it/grazzini`. To extract it, simply use `tar -xzvf dynnlo-v1.5.tgz` and the `dynnlo-v1.5` directory will be created. The structure of the directory is

- `bin`: The directory containing the executable `dynnlo` and the input and output files.
- `doc`: The directory containing this note.
- `obj`: The directory containing the object files.
- `src`: The directory containing the source of the code.

2 Input parameters

In our calculation, the W and Z bosons are treated off shell, thus including finite-width effects, and their leptonic decay retains the corresponding spin correlations. The mass and total width of the W boson are $m_W = 80.385$ GeV and $\Gamma_W = 2.085$ GeV. The mass and total width of the Z boson are $m_Z = 91.1876$ GeV and $\Gamma_Z = 2.4952$ GeV. As for the electroweak couplings, we use the so called G_μ scheme, where the input parameters are G_F , m_Z, m_W. The Fermi constant is set to the value $G_F = 1.1663787 \times 10^{-5}$ GeV^{-2}, and we use the following (unitarity constrained) values of the CKM matrix elements: $V_{ud} = 0.97427$, $V_{us} = 0.2253$, $V_{ub} = 0.00351$, $V_{cd} = 0.2252$, $V_{cs} = 0.97344$, $V_{cb} = 0.0412$. All these values of EW parameters are taken from the PDG 2012 [2]. The EW couplings of the W and Z bosons to quarks and leptons are treated at the tree level, so that the above parameters are sufficient to fully specify the EW content of our calculation.

3 Implementation of cuts

Before compiling the program, the user must choose the cuts to apply on the final state. This is done through the `cuts` and `isolation` functions. The default version of the function `cuts` contains selection cuts that are typically used in the experimental analysis. In the default version, the lines of the code implementing the various cuts are all commented, thus the program will return the total cross section for the selected process in the narrow width approximation. To take into account finite width effect the logical variable `zerowidth` should be set to `false` in the input file [1]. To activate the various cuts the user must uncomment the corresponding lines in `cuts.f`. Lepton isolation can be implemented by switching to `true` the logical variable `isol`. The parameters to define the isolation procedure are set in the `isolation` function. The program makes possible the identification of final-state jets, in addition to the leptons. Jets [5] are reconstructed according to the k_T algorithm with $R = 0.4$. Different jet algorithms (as the *anti-*k_T or the *cone*) and different values of R can be implemented by modifying the `setup` subroutine.

The files `cuts.f` and `isolation.f` can be found in the `/src/User` directory. The `setup.f` source file can be found in the `/src/Need` directory.

4 Compilation

The program is self-contained and it has been successfully tested on Linux and Mac-OS X environments. To compile the code, descend in the `dynnlo` directory and simply type

- `make`

To run it go in the `bin` directory and type:

- `dynnlo < infile`

5 The input file

This is a typical example of input file:

```
8d3                ! sroot
```

[1] Note that with `zerowidth=.false.` a hard scale must be set, to have a cross section that is computable in QCD perturbation theory. The hard scale is set by applying cuts on the leptons. These cuts must necessarily enforce a lower limit on the invariant mass of the vector boson (lepton pair), and the mass scale of this lower limit must be in the perturbative region.

```
1 1                ! ih1, ih2
1                  ! nproc
80.385d0 80.385d0 ! mur, muf
2                  ! order
'tota'             ! part
.false.            ! zerowidth
50d0 7d3           ! mwmin, mwmax
15 8000000         ! itmx1, ncall1
30 8000000         ! itmx2, ncall2
617                ! rseed
92 0               ! set, member (native PDFs)
'MSTW2008lo68cl.LHgrid' 0 !  set, member (LHAPDFs)
'nnlo'             ! runstring
```

- `sroot`: Double precision variable for centre–of–mass energy (GeV).
- `ih1,ih2`: Integers identifying the beam (proton=1, antiproton=-1).
- `nproc`: Integer identifying the process: $W^+ \to l^+\nu$ (`nproc=1`), $W^- \to l^-\bar{\nu}$ (`nproc=2`), $Z/\gamma^* \to l^+l^-$ (`nproc=3`).
- `mur, muf`: Renormalization (μ_R) and factorization (μ_F) scales (GeV): can be be different from each other but always of the order of m_W or m_Z (or $m_{l\nu}$ and m_{ll} if these invariant masses are very different from m_W and m_Z). By switching to `true` the logical variable `dynamicscale` in the `setup` subroutine, μ_R and μ_F are set equal to $m_{l\nu}$ or m_{ll}.
- `order`: Integer setting the order of the calculation: LO (0), NLO (1), NNLO (2).
- `part`: String identifying the part of the calculation to be performed: `virt` for virtual contribution, `real` for real contribution, `tota` for the complete calculation.
- `itmx1, ncall1`: Number of iterations and calls to VEGAS for setting the grid.
- `zerowidth`: When this logical variable is set to `true` the vector bosons are produced on-shell. To take into account finite width effect this variable should be set to `false`.
- `mwmin, mwmax`: Lower and higher limit on the vector boson (lepton pair) invariant mass.
- `itmx2, ncall2`: Number of iterations and calls to VEGAS for the main run.
- `rseed`: Random number seed.
- `iset, nset`: Integers identifying respectively the PDF set chosen and the member for PDF errors (if the native PDF interface is used). A list of available PDFs is given below.
- `PDFname, PDFmember`: String identifying the PDF set chosen and integer identifying the member for PDF errors (if the LHAPDF interface is used).
- `runstring`: String for grid and output files.

6 Output

At the end of the run, the program returns the cross section and its error. The program also writes an output file in the topdrawer format containing the required histograms with an estimate of the corresponding statistical errors. During the run, the user can control the intermediate results. The plots are defined in the file `/src/User/plotter.f`. The user can easily modify this subroutine according to his/her needs.

The NLO computation of a cross section with accuracy at the percent level typically requires a few hours of run on a standard PC (say a Pentium 4 at 3 GHz). At NNLO the required run time is between one and two days. To obtain smooth distributions, this estimated run time should be multiplied by a factor of two.

7 Parton distributions

The `DYNNLO` program can be compiled with its own Parton Distribution Functions (PDF) interface (set `PDFROUTINES = NATIVE` in the `makefile`) or with the LHAPDF interface (set `PDFROUTINES = LHAPDF` in the `makefile`). We point out that the value of $\alpha_S(m_Z)$ is not adjustable; it is hard-wired with the value of $\alpha_S(m_Z)$ in the parton distributions. Moreover, the choice of the parton distributions also specifies the number of loops that should be used in the running of α_S. A list of available parton densities for the native PDF interface is given in Table 1.

When dealing with PDF uncertainties, the `nset` variable is used to distinguish the PDF grids corresponding to different eigenvectors. When CTEQ6.6M partons are used, `nset` varies in the range `nset=0,44`. When NNPDF2.0_100 partons are used, `nset` varies in the range `nset=0,100`. When MSTW2008 partons are used, `nset` varies in the range `nset=0,40` and the uncertainties we consider are those at 68% CL. When GJR08VF NLO or JR09VF NNLO partons are used, the `nset` variable should be in the range `nset=-13,13`. For A06 (ABKM09) NNLO partons, the variable `nset` should vary in the range `nset=0,23` (`nset=0,25`). The default choice is `nset=0`, corresponding to the central set. The variable `nset` is dummy when other PDF sets are used.

8 From version 1.1 to version 1.2

The main change done in version 1.2 with respect to version 1.1 is that the evaluation of the subtraction counterterm is made substantially faster. As a consequence, the speed of the code improves by at least a factor 1.5. A minor bug has been corrected in the dynamic scale settings, whose effect is however negligible.

9 From version 1.2 to version 1.3

The only change done in version 1.3 with respect to version 1.2 is that a bug affecting the program when running with `dynamicscale=.false.` has been fixed.

10 From version 1.3 to version 1.4

In version 1.4 some refinements in the subtraction procedure have been implemented and two bugs affecting the real and the virtual contribution has been fixed.

11 From version 1.4 to version 1.5

In version 1.5 a bug in the factorization scale dependence has been corrected.

References

[1] S. Catani, L. Cieri, G. Ferrera, D. de Florian and M. Grazzini, Phys. Rev. Lett. **103** (2009) 082001; S. Catani and M. Grazzini, Phys. Rev. Lett. **98** (2007) 222002.

[2] J. Beringer *et al.* [Particle Data Group Collaboration], Phys. Rev. D **86** (2012) 010001.

[3] S. Catani and M. H. Seymour, Phys. Lett. B **378** (1996) 287, Nucl. Phys. B **485** (1997) 291 [Erratum-ibid. B **510** (1998) 503].

[4] J. Campbell, R.K. Ellis, *MCFM - Monte Carlo for FeMtobarn processes*, `http://mcfm.fnal.gov`

[5] S. Catani, Y. L. Dokshitzer, M. H. Seymour and B. R. Webber, Nucl. Phys. B **406** (1993) 187; S. D. Ellis and D. E. Soper, Phys. Rev. D **48** (1993) 3160.

iset	Pdf set	$\alpha_{\rm S}(M_Z)$
1	CTEQ4 LO	0.132
2	CTEQ4 Standard NLO	0.116
11	MRST98 NLO central gluon	0.1175
12	MRST98 NLO higher gluon	0.1175
13	MRST98 NLO lower gluon	0.1175
14	MRST98 NLO lower $\alpha_{\rm S}$	0.1125
15	MRST98 NLO higher $\alpha_{\rm S}$	0.1225
16	MRST98 LO	0.125
21	CTEQ5M NLO Standard Msbar	0.118
22	CTEQ5D NLO DIS	0.118
23	CTEQ5L LO	0.127
24	CTEQ5HJ NLO Large-x gluon enhanced	0.118
25	CTEQ5HQ NLO Heavy Quark	0.118
28	CTEQ5M1 NLO Improved	0.118
29	CTEQ5HQ1 NLO Improved	0.118
30	MRST99 NLO	0.1175
31	MRST99 higher gluon	0.1175
32	MRST99 lower gluon	0.1175
33	MRST99 lower $\alpha_{\rm S}$	0.1125
34	MRST99 higher $\alpha_{\rm S}$	0.1225
41	MRST2001 NLO central gluon	0.119
42	MRST2001 NLO lower $\alpha_{\rm S}$	0.117
43	MRST2001 NLO higher $\alpha_{\rm S}$	0.121
44	MRST2001 NLO better fit to jet data	0.121
45	MRST2001 NNLO	0.1155
46	MRST2001 NNLO fast evolution	0.1155
47	MRST2001 NNLO slow evolution	0.1155
48	MRST2001 NNLO better fit to jet data	0.1180
51	CTEQ6L LO	0.118
52	CTEQ6L1 LO	0.130
53	CTEQ6M NLO	0.118
55	CTEQ6.6M NLO	0.118
49	MRST2002 LO	0.130
61	MRST2002 NLO	0.1197
62	MRST2002 NNLO	0.1154
65	GJR08VF LO	0.1263
66	GJR08VF NLO	0.1145
67	JR09VF NNLO	0.1124
71	MRST2004 NLO	0.1205
72	MRST2004 NNLO	0.1167
75	A06 NNLO	0.1128
80	NNPDF2.0_100 NLO	0.1190
85	ABKM09 NNLO	0.1129
90	MSTW2008 LO	0.13939
91	MSTW2008 NLO	0.12018
92	MSTW2008 NNLO	0.11707

Table 1: Available pdf sets and their corresponding `iset` and values of $\alpha_S(M_Z)$.