Mass effects

by K. Hamilton, P. Nason and G. Zanderighi

This manual documents the upgrade of the HJ-MiNLO code illustrated in ref. [1] [2] [3], where we describe the implementation of quark-mass effects in the HJ-MiNLO generator, and its NNLO accurate reweighting. It also updates the setup found in HJ/NNLOPS, describings the reweighting of the HJ-MiNLO ouptut in order to achieve NNLO accuracy, since in the new code we use extensively the reweighting conventions agreed upon in the Les Houches convention described in http://phystev.in2p3.fr/wiki/2013:groups:tools_lheextension. The new NNLOPS code is now in HJ/NNLOPS-mass-effects and in the main HJ directory. The HJ/NNLOPS directory has not been modified, and the manual included there is still useful for all aspects that have not been updated in the present manual.

The use of the code is illustrated in the directory HJ/PaperRun, where a script for reproducing the runs performed for the preparation of ref. [1] is given. Here we refer to this example in order to illustrate how the program works.

1 Production of the HJ-MiNLO events with mass effects.

The inclusion of mass effects is controlled by the following flags in the powheg.input file:

```
quarkmasseffects 1   ! Include quark mass effects
topmass 172.5d0      ! top quark mass (172.5 in HNNLO-patches/mdata.f)
bottommass 3.38d0    ! bottom quark mass in MSbar at MH
bmass_in_minlo 1     ! Include quark mass effects in Sudakov exponent
```

The first line must be present if we want to include quark mass effects. If the second line is present, top mass effects are included. If the third line is present, also bottom mass effects are included. If the fourth line is present (and also the third line is present), bottom mass effects are assumed to affect also the MiNLO Sudakov form factor, as discussed in [1].

In the setup of the PaperRun directory, a first sample is generated with no quark mass effects. After that, the mass effects when only top is included, when both top and bottom are included, and when both top and bottom are included, including the effects of the bottom mass in the Sudakov form factor, are all added using the POWHEG reweighting facility.

The $m_t = \infty$ sample is generated with the following Les Houches labels:

```
lhrwgt_group_name 'MiNLO'
lhrwgt_id 'mtinf'
lhrwgt_descr 'mt infinity result'
```

and no quark-mass flags set. This results in Les Houches event files including in the header the follwing block:

```
<header>
<initrwgt>
<weightgroup name='MiNLO'>
<weight id='mtinf'> mt infinity result </weight>
</weightgroup>
</initrwgt>
</header>
```

The program is then run in reweighting mode, with the following variables settings:

```
compute_rwgt 1
quarkmasseffects 1   ! Include quark mass effects
topmass 172.5d0      ! top quark mass (172.5 in HNNLO-patches/mdata.f)
lhrwgt_group_name 'MiNLO'
lhrwgt_id 'mt'
lhrwgt_descr 'Finite mt only'
```

This run takes as input the name of an existing Les Houches event file, <file>.lhe, and creates a new file, named <file>-rwgt.lhe, including the new weight. This file is then renamed <file>.lhe, and the procedure is repeated, with the following settings:

```
compute_rwgt 1
quarkmasseffects 1   ! Include quark mass effects
topmass 172.5d0      ! top quark mass (172.5 in HNNLO-patches/mdata.f)
bottommass 3.38d0    ! bottom quark mass in MSbar at MH
lhrwgt_group_name 'MiNLO'
lhrwgt_id 'mtmb'
lhrwgt_descr 'Finite mt and mb'
```

and then again with the settings:

```
compute_rwgt 1
quarkmasseffects 1   ! Include quark mass effects
topmass 172.5d0      ! top quark mass (172.5 in HNNLO-patches/mdata.f)
bottommass 3.38d0    ! bottom quark mass in MSbar at MH
bmass_in_minlo 1
lhrwgt_group_name 'MiNLO'
lhrwgt_id 'mtmb'
lhrwgt_descr 'Finite mt and mb, b mass in minlo'
```

At the end of the procedure, the Les Houches file carries the following reweighting information in the header:

```
<header>
<initrwgt>
<weightgroup name='MiNLO'>
<weight id='mtinf'> mt infinity result </weight>
<weight id='mt'> Finite mt only </weight>
<weight id='mtmb'> Finite mt and mb </weight>
<weight id='mtmb-bminlo'> Finite mt and mb, b mass in minlo </weight>
</weightgroup>
</initrwgt>
</header>
```

and after each event, we find reweighting information of the form:

```
<rwgt>
<wgt id='mtinf'>  0.141990268E+02 </wgt>
<wgt id='mt'>  0.838351860E+01 </wgt>
<wgt id='mtmb'>  0.846010435E+01 </wgt>
<wgt id='mtmb-bminlo'>  0.845939328E+01 </wgt>
</rwgt>
```

2 Generation of the `hnnlo` output for reweighting

The fundamental NNLO ingredient required to obtain NNLO accurate event samples is, for gluon fusion Higgs production, the Higgs boson's rapidity spectrum.[1] This is obtained using the `hnnlo` program of [4] [5] [6]. We here give instructions on how to obtain such distributions, suitable for combination with the `HJ-MiNLO` events via the `nllopsreweighter` program.

1. Make sure that the `lhapdf` package is installed:
   https://lhapdf.hepforge.org

2. Download `hnnlo` from the url:
   `$ wget` http://theory.fi.infn.it/grazzini/codes/hnnlo-v2.0.tgz

3. Unpack the tarball in a convenient location:

   ```
   $ tar -xzvf hnnlo-v2.0.tgz
   $ ls hnnlo-v2.0
   ```

   Under the parent directory `hnnlo-v2.0` one should find subdirectories `bin`, `doc`, `obj`, `src`, and a `makefile`.

4. Enter the `hnnlo` parent directory
   ```
   $ cd hnnlo-v2.0
   ```

5. Replace the `hnnlo` default `makefile` with the one from the `HJ/NNLOPS-mass-effects` directory
   ```
   $ cp /path/to/HJ/NNLOPS-mass-effects/HNNLO-makefile ./makefile
   ```

6. Copy the `hnnlo` patches directory into the parent directory
   ```
   $ cp -r /path/to/HJ/NNLOPS-mass-effects/HNNLO-patches ./
   ```

7. Build the code
   ```
   $ make
   ```
   A message `--> HNNLO compiled with LHAPDF routines <--` indicates success.

8. Copy over the required input file to the `bin` directory. In the `PaperRun` example:

   ```
   $ cd bin
   $ cp ...POWHEG-BOX-V2/HJ/PaperRun/HNNLO*.input .
   ```

9. Perform the run. For example:

   ```
   $ ./hnnlo < HNNLO-LHC8-R04-APX0-11.input > my.log
   ```

This generates the `HNNLO-LHC8-R04-APX0-11-10103.top` file. The file name reminds us that the run was performed with central renormalization and factorization scales (the 11 string), `approxim 0`, and `rseed 01103`. In order to obtain the `HNNLO-LHC8-R04-APX0-11.top` file, 350 runs were performed. Each run takes of the order of 20 hours. The large number of runs is needed if one wants to get precise results also near the end-points of the rapidity spectrum.

The output of the different runs were combined using the `mergedata.f` program (located in the `HJ/NNLOPS` directory). It is used as follows:

```
$ gfortran -o mergedata nergedata.f
$ ./mergedata
```

1. In general, for NNLOPS simulations one requires an NNLO accurate distribution for some Born kinematics. In this case the Born kinematics is fully specified in terms of the rapidity of the boson.

```
1
filename1
filename2
...
```

where the list of filenames is terminated by an empty line.

3 Generation of NNLOPS reweighted events

In order to generate an NNLOPS sample, one needs first to compute certain distributions using the hnnlo program of [4] [5] [6]. This is documented in the HJ/NNLOPS/hnnlops.pdf document. In the `PaperRun` directory, there are three input files and three output files of the `hnnlo` program, with matching names. They correspond to the `hnnlo` runs for the $m_t = \infty$, m_t only, and m_t, m_b effects included. The data files (ending in `.top`) are needed, together with the `.lhe` file generated as described in the previous section, for the generation of the NNLOPS weights. These are obtained with the `nnlopsreweighter` program, that is generated in the `HJ` directory by simply doing

```
$ make nnlopsreweighter
```

The `nnlopsreweighter` program reads the input file `nnlopsreweighter.input`, that contains the following lines:

```
lhfile pwgevents.lhe
```

that instructs the program to read the events from the `pwgevents.lhe` file. Then the section

```
nnlofiles
'nn-mtinf' HNNLO-LHC8-R04-APX0-11.top
'nn-mt' HNNLO-LHC8-R04-APX1-11.top
'nn-mtmb' HNNLO-LHC8-R04-APX2-11.top
```

specify the `hnnlo` output file that should be read. Finally, we have the section

```
<initrwgt>
<weightgroup name='nnl'>
<weight id='nnlops-mt'> combine 'nn-mtinf' and 'mtinf' </weight>
<weight id='nnlops-mt'> combine 'nn-mt' and 'mt' </weight>
<weight id='nnlops-mtmb'> combine 'nn-mtmb' and 'mtmb' </weight>
<weight id='nnlops-mtmb-bminlo'> combine 'nn-mtmb' and 'mtmb-bminlo' </weight>
</weightgroup>
</initrwgt>
```

This has the format of a Les Houches header for reweighting. By running the `nnlopsreweighter` program, we produce a new event file, named `pwgevents.lhe.nnlo`, containing the following header

```
<header>
<initrwgt>
<weightgroup name='MiNLO'>
<weight id='mtinf'> mt infinity result </weight>
<weight id='mt'> Finite mt only </weight>
<weight id='mtmb'> Finite mt and mb </weight>
<weight id='mtmb-bminlo'> Finite mt and mb, b mass in minlo </weight>
</weightgroup>

<weightgroup name='nnl'>
<weight id='nnlops-mt'> combines 'nn-mtinf' and 'mtinf' </weight>
```

```
<weight id='nnlops-mt'> combines 'nn-mt' and 'mt' </weight>
<weight id='nnlops-mtmb'> combines 'nn-mtmb' and 'mtmb' </weight>
<weight id='nnlops-mtmb-bminlo'> combines 'nn-mtmb' and 'mtmb-bminlo' </
weight>
</weightgroup>

</initrwgt>
</header>
```

where the new entry mentioned in the `nnlopsreweighter.input` appear, as well as the corresponding weights at the end of each event. The weights are obtained as follows. The `nnlopsreweighter` program looks in the description line of each new weight mentioned in the `nnlopsreweighter.input` file, looking for a string that appears as a weight id in the `pwgevents.lhe` file, and for a string that appears as a prefix to a file name in the `nnlofiles` section of the `nnlopsreweighter.input` file. The new weight is then defined as the combination of the corresponding weight in the event file, and in the corresponding `hnnlo` output file. In this way, any combination can be produced, and it is up to the user to use a sensible one. In the example reported in the `PaperRun` directory, we combine `HJ-MiNLO` and `hnnlo` results that match in the accuracy of the quark mass effects that have been included. Note that the `'mtmb'` and the `'mtmb-bminlo'` weights are reweighted with the same `'nn-mtmb'` tagged file, i.e. the `HNNLO-LHC8-R04-APX2-11.top` file. In fact, there is no analog of the `bmass_in_minlo` option in the `hnnlo` program, that we only use to compute p_T integrated distributions, without thus including any Sudakov form factor.

Bibliography

[1] K. Hamilton, P. Nason, and G. Zanderighi, *Finite quark-mass effects in the NNLOPS POWHEG+MiNLO Higgs generator*, xxxx.xxxx.

[2] K. Hamilton, P. Nason, E. Re, and G. Zanderighi, *NNLOPS simulation of Higgs boson production*, *JHEP* **1310** (2013) 222, [1309.0017].

[3] K. Hamilton, P. Nason, C. Oleari, and G. Zanderighi, *Merging H/W/Z + 0 and 1 jet at NLO with no merging scale: a path to parton shower + NNLO matching*, 1212.4504.

[4] M. Grazzini, *NNLO predictions for the Higgs boson signal in the $H \to W^+W^- \to \bar{l}\nu\, l\bar{\nu}$ and $H \to ZZ \to 4l$ decay channels*, *JHEP* **0802** (2008) 043, [0801.3232].

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

[6] M. Grazzini and H. Sargsyan, *Heavy-quark mass effects in Higgs boson production at the LHC*, *JHEP* **1309** (2013) 129, [1306.4581].