DRAFT: RESUMMINO Manual

This draft is meant to help people install and use RESUMMINO 2.0.1 with LOOPTOOLS. Please, ask any question to help us improve this walkthrough for RESUMMINO by sending emails to mpasunder@uni-muenster.de.

Some things such as a description of the output and the significance of certain numbers that are found in the output are not treated in this draft manual. I will only include them if there is at least one user that has some need of it.

Installation

Before installing the RESUMMINO code one requires the following prerequisites, namely: some BOOST headers, an older version of the GNU Scientific Library and the LHAPDF library (the somewhat tricky LOOPTOOLS installation is done automatically.

the BOOST headers

On RHEL/Scientific Linux or Fedora one uses:

```
sudo yum install boost boost-devel
```

On Debian/Ubuntu one uses:

```
sudo apt-get install libboost-dev
```

On MacOS one either uses MacPorts and types:

```
sudo port install boost
```

or one uses Homebrew and does:

```
brew install boost
```

the GNU Scientific Library

We do not use the standard methods for a system-wide installation of `gsl` because RESUMMINO requires an older version of `gsl`. We first download the `gsl-1.16` version into (optionally) the `download` folder. We then unpack this file into (optionally) our `lib` folder. We then `./configure`, `make` and `make install` our `gsl`, after which we clean up the now useless directory from which it was configured.

```
wget http://mirror.netcologne.de/gnu/gsl/gsl-1.16.tar.gz  ...
... -P downloads
tar -xvf ./downloads/gsl-1.16.tar.gz -C ./lib
cd lib/gsl-1.16
./configure --prefix=/full/path/to/lib/gsl
make ; make install
rm -r gsl-1.16
```

the LHAPDF Library

The installation for the LHAPDF library is very similar, and if you already have it on your system you can skip the installation and should just remember the location of the `lhapdf` directory for RESUMMINO's installation. The environment variables that I suggest are merely there to ensure you have fully and correctly installed the LHAPDF library, they are not so important for the installation of RESUMMINO.

```
wget http://www.hepforge.org/archive/lhapdf/LHAPDF-6.1.6.tar.gz
... -P downloads
tar -xvf ./downloads/LHAPDF-6.1.6.tar.gz -C ./lib
cd lib/LHAPDF-6.1.6
./configure --prefix=/full/path/to/lib/lhapdf
make ; make install
cd ../ ; rm -r LHAPDF-6.1.6
```

Now to allow use of LHAPDF by other codes and by your `python-2.7` installation, you should add the following lines to your `~/.bashrc`$^{\text{UNIX}}$ or `~/.bash_profile`$^{\text{MacOS}}$

```
...
export LHAPDF_DIR=/full/path/to/lib/lhapdf
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$LHAPDF_DIR/lib
export LHAPDF_DATA_PATH=$LHAPDF_DIR/share/LHAPDF
export PYTHONPATH=$PYTHONPATH:$LHAPDF_DIR/python2.7/site-packages
...
```

Before one can use the LHAPDF library one needs to download the PDF sets. This is done for any PDF set such as `cteq66` by executing the following lines:

```
$LHAPDF_DIR/bin/lhapdf install cteq66
```

the Resummino code

The installation of Resummino is similar to the previous installations.

```
wget http://pauli.uni-muenster.de/~masunder/downloads/...
...resummino/resummino-2.0.1-looptools.tar.bz2 -P downloads
tar -xvf ./downloads/resummino-2.0.1-looptools.tar.bz2 -C ./lib
cd lib/resummino-2.0.1
cmake ./ -DLHAPDF=/full/path/to/lib/lhapdf ...
... -DCMAKE_PREFIX_PATH=/full/path/to/lib/gsl
make ; make install
```

Usage of Resummino

First of all, congratulations - if you are reading this apparently the installation was successful. The next step is to use Resummino for your process of interest with your preferred setup. By modifying `./input/resummino.in` and by using the optional arguments as described in the following, one can accomplish this.

the Input Card

Here we go through the most important optional arguments that can be set in the `./input/resummino.in` input card that can be found in `/full/path/to/lib/resummino-2.0.1`.

```
# Input card for Resummino.

# This input card defines the hadron collider parameters and the
# process. The SUSY model is defined in a separate file in the
# SLHA file format.
# All energies and masses are in GeV.

# Collider parameters.
collider_type         = proton-proton  # proton-proton
                                       # or proton-antiproton
center_of_mass_energy = 13000
```

We consider a process at the LHC, which collides protons with each other. Unlike the Tevatron, which collided protons with anti-protons. The LHC currently operates at it's designed center of mass energy of 13 GeV but for older datasets one could change this to 8 GeV.

```
# Outgoing particles using the value of the PDG-scheme number.
# The particles are listed in the table below. A minus in front
# of the PDG means charge conjugation.
```

```
# Note: Only total cross section at LO possible for associated
# squark-gaugino production!
# ------------------------------------------------------------------
# | ~e_L-      = 1000011 | ~e_R-    = 2000011 | ~nu_eL  = 1000012 |
# | ~mu_L-     = 1000013 | ~mu_R-   = 2000013 | ~nu_muL = 1000014 |
# | ~tau_1-    = 1000015 | ~tau_2-  = 2000015 | ~nu_tau = 1000016 |
# ------------------------------------------------------------------
# | ~chi_10    = 1000022 | ~chi_20 = 1000023 | ~chi_30 = 1000025 |
# | ~chi_40    = 1000035 | ~chi_1+ = 1000024 | ~chi_2+ = 1000037 |
# ------------------------------------------------------------------
# | ~u_L       = 1000002 | ~u_R     = 2000002 | ~d_L     = 1000001 |
# | ~d_R       = 2000001 | ~c_L     = 1000004 | ~c_R     = 2000004 |
# | ~s_L       = 1000003 | ~s_R     = 2000003 | ~t_L     = 1000006 |
# | ~t_R       = 2000006 | ~b_L     = 1000005 | ~b_R     = 2000005 |
# ------------------------------------------------------------------
# | ~g         = 1000021 |                                         |
# ------------------------------------------------------------------
#
particle1 =  1000022
particle2 =  1000023
```

The above specifications for `particle1` and `particle2` allows us to look at hadronic production of two neutralino's, namely χ_1^0 and χ_2^0.

```
# Defines the computation to be performed. Three computations are
# supported:
#
# - result = total: Outputs the total cross section.
#
# - result = pt:    Outputs the value for the transverse momentum
#                   at the value specified by the 'pt' variable.
#
# - result = ptj:   Outputs the value for the transverse momentum
#                   at the value specified by the 'pt' variable
#                   using the joint resummation formalism.
#
# - result = m:     Outputs the value for the invariant mass
#                   distribution at the value specified by the
#                   'M' variable.
#
result = total  # total, pt, ptj or m.
M      = auto   # auto = sqrt((p1 + p2)^2)
pt     = auto

# SLHA input file that defines the SUSY benchmark point.
slha = slha.in
```

Here, one specifies what it is that one wishes to compute e.g. the total cross-section, the value for the transverse momentum or the invariant mass distribution. More importantly, though the model parameter are specified here through a reference to `slha.in`. One can use this or a different SUSY benchmark point by putting a file in the SUSY Les Houches Accord (SLHA) format into the `./input` directory with a funny name such as `hino.dat` and then put into `./input/resummino.in`: `slha = hino.dat`.

```
# PDF sets for LO and NLO. They should be present in the LHAPDF
# local setup.
pdf_format = lhpdf  # lhgrid or lhpdf
pdf_lo     = MSTW2008lo90cl
pdfset_lo  = 0
pdf_nlo     = MSTW2008nlo90cl
pdfset_nlo = 0

#scale factors
# (1.0 is central scale mu = (m1 + m2) / 2)
mu_f = 1.0
mu_r = 1.0
```

One needs to specify which PDF set to use and ensure that it is installed in `$LHAPDF_DATA_PATH`. If not, use `$LHAPDF_DIR/bin/lhapdf install MSTW2008lo90cl` to install it as specified earlier. Also, make sure that you specify a leading order PDF set for the leading order calculation and a next-to-leading order PDF set for the nex-to-leading order computation.

When you want to compute PDF variation and scale variation errors, then one should loop over all members of the PDF set and over all required factorisation and renormalisation scales. In other words, one should set `pdfset_lo` and `pdfset_nlo` to all numbers between `pdfset_(n)lo=0` and `pdfset_(n)lo=40` and run RESUMMINO for each. For the central scale i.o.w. `pdfset_(n)lo=0`, one then varies `{ mu_f , mu_r } = {{ 0.5 , 0.5 }, ..., { 1.0 , 0.5 }, ..., { 2.0 , 2.0 }}`

```
# Integration parameters.
precision = 0.005 # desired precision
max_iters = 5  # maximum iterations

# optional PDF fit parameter
# weights
# (If you get weird fit results decrease the weight up to -2.0)
weight_valence = -1.0
weight_sea = -1.0
weight_gluon = -1.0
# fit PDF from xmin to 1 (auto = mis/sh)
xmin = auto
```

If the PDF fit does not converge or if the precision is not good enough one can fiddle around with these parameters. I would suggest one first tries to increase the number of iterations and change the desired precision before one attempts changing the PDF fit-weights.

Running the RESUMMINO code

After a successful specification of the SUSY benchmark point (**fully** filled-out* in SLHA format) And after a sensible filled out input card, having gone over all

*So, upon executing RESUMMINO with its input card no unfilled/replaced *options* such as `@a` are allowed in the SLHA-file.

options one will want to do RESUMMINO run.

The general command with some or none optional arguments denoted by `OPTIONS` to run RESUMMINO is given by:

```
./bin/resummino  [OPTIONS] ./input/resummino.in
```

The most important optional arguments are given by:

- `--lo` stops the calculation after the LO result.
- `--nlo` stops the calculation after the NLO result.
- `--nnll` computes ordinary (not collinear improved) threshold resummation at the NLO+NNLL accuracy level for Drell-Yan like processes.
- `--parameter-log=params.log` stores the values of all parameters, masses and couplings in a log file `params.log` for future reference.

The RESUMMINO code accepts some additional optional arguments to facilitate the running of RESUMMINO in the terminal without having to continuously modify the input card.

- `--output=/path/to/output.dat` outputs a summary of the result of the computation in the `JSON`-format.
- `--pdfset_nlo=cteq66` to alter the PDF set member to any of the error PDF sets for example.
- `--mu_r=0.5` to alter the renormalisation scale without touching the input card.
- `--mu_f=2.0` to alter the factorisation scale without touching the input card.