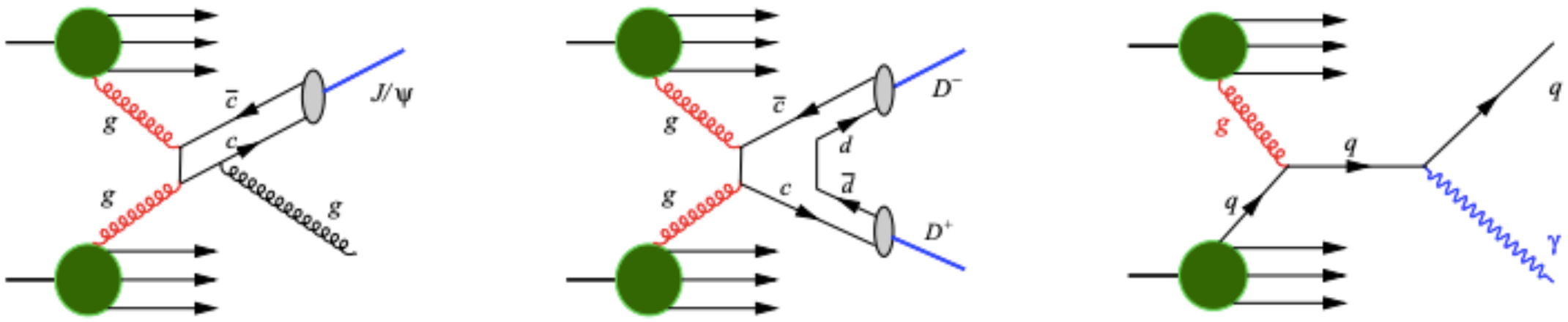
Study of charmonia production at SPD with HELAC-Onia

Gridin Andrei (JINR)
andreigridin@jinr.ru

SPD Physics and MC meeting, 20.03.2024



At NICA SPD we are going to study gluon content of proton and deuteron. In this case it might be interesting to include the polarization of final state charmonia into Monte Carlo.

HELAC-Onia is a programm for automated computation of heavy-quarkonium helicity amplitudes within non-relativistic QCD framework.

- Developed by Jean Philipp Lansberg group.
- Widely used by LHC experiments for validation of charmonia production models (CS, CO, NRQCD LO/NLO) for single/double/triple charmonia production.

https://helac-phegas.web.cern.ch/helac-onia.html

Hua-Sheng Shao
HELAC-Onia 2.0: an upgraded matrix-element and event generator for heavy quarkonium physics
arXiv:1507.03435v1 [hep-ph]

Features

- Works in fixed target and collider modes;
- Colliding particles:
  pp, $p\bar{p}$, e^+e^-, $e^{\pm}p$, $e^{\pm}\bar{p}$, γp, $\gamma\bar{p}$, $\gamma\gamma$
- Supports polarization of the final states: $J/\psi \to \mu\mu, \chi_{C_{1,2}} \to J/\psi\gamma$
- DPS, MPI;
- Open heavy flavor meson pair production;

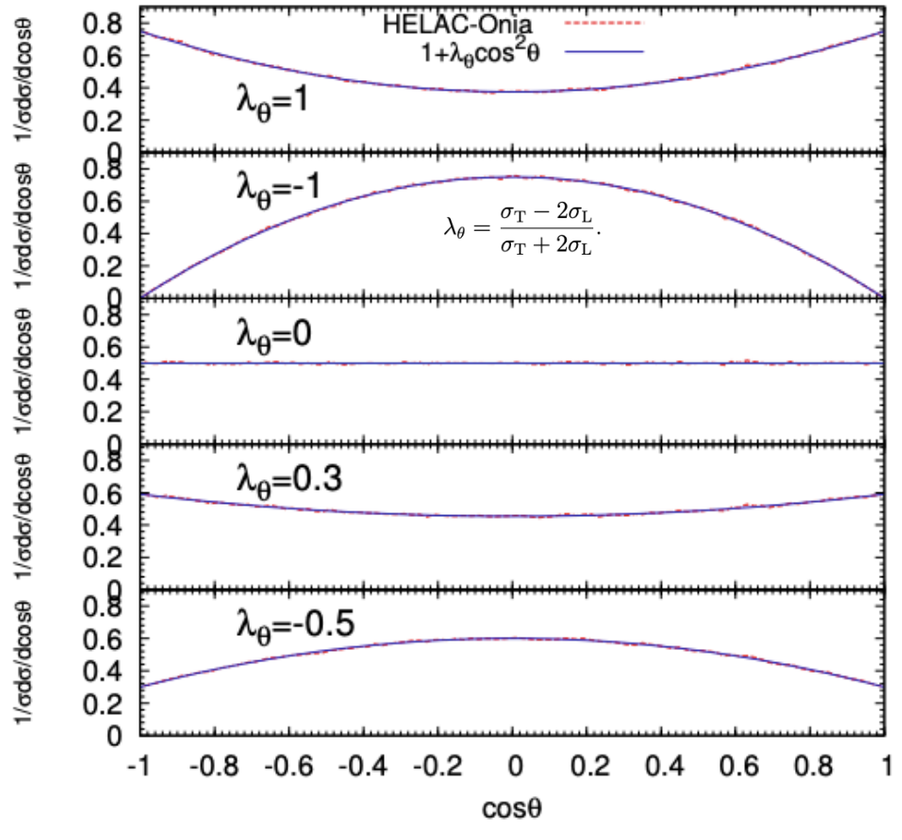


Figure 4: Validation of lepton angular distributions in $J/\psi \to \ell^+\ell^-$.

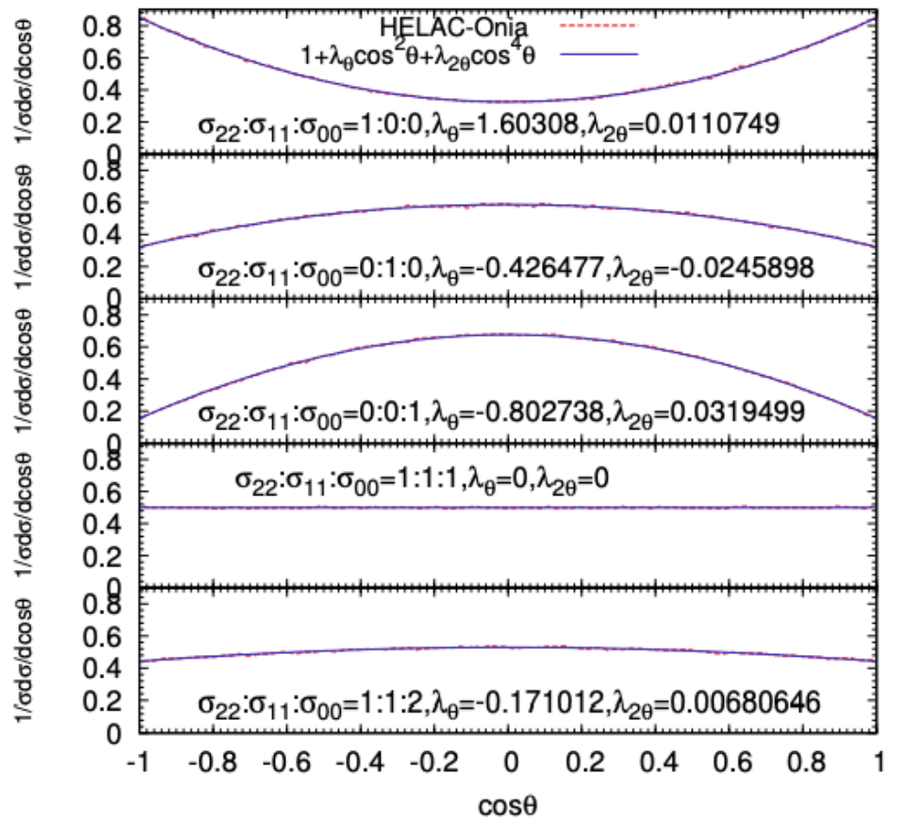


Figure 6: Validation of J/ψ angular distributions in $\chi_{c2} \to J/\psi + \gamma$.

1) Build the HELAC-Onia

- Download the generator. The latest version is available at /afs/cern.ch/work/h/hshao/public/HELAC-Onia-2.5.5.tar.gz
- In Helac-Onia/input/ho_configuration.txt provide paths to additional libraries (ROOT, LHAPDF, Pythia8, etc).
- Run configuration script ./config in the generator directory
- After successful compilation binaries will appear in Helac-Onia/bin directory

2) Set MC settings

In Helac-Onia/input/ there are:
default.inp — basic settings, that are overwritten be user settings;
user.inp — user defined settings;

```
exp3pjQ   F          # open (T) or not open (F) explici 3pj modes,if opened,returns 3pj (j=0,1,2) seperately.
modes 0              # modes 0: unpolarized mode 1: polarization mode; When modes=1, the exp3pjQ for
                     # 3pj1 polarization must be set True.
LSJ   1              # SDME1,SDME2 represent 0:L,1:S,2:J
SDME1 0              # The first index for the Spin Density Matrix Element of the first quarkonium
SDME2 0              # The second index for the Spin Density Matrix Element of the first quarkonium
PolarFrame 1         # 1: Helicity Frame 2: Collins-Soper Frame 3:Gottfried-Jackson frame 4:Target frame
```

decay_user.inp — set decay channels;

ho_configuration.txt — configuration paths used for compilation

Demo programs also have **polarization.inp** where one can provide the value of λ_θ

3) Run the generator

1) One can directly run the generator:
./ho_cluster

```
=============================================================
=                                                           =
=                     H E L A C - O n i a                   =
=                                                           =
=        VERSION 2.5.5                 2020-05-15           =
=                                                           =
=                                                           =
=      Author:         Hua-Sheng Shao                       =
=   Affliation:             LPTHE                           =
=     Contact:       huasheng.shao@lpthe.jussieu.fr         =
=    Reference:    Comput.Phys.Commun. 184 (2013) 2562       =
=        arXiv:            1212.5293                        =
=    Reference:    Comput.Phys.Commun. 198 (2016) 238        =
=        arXiv:            1507.03435                       =
=    Webpage: http://helac-phegas.web.cern.ch/helac-phegas  =
=                                                           =
=============================================================
INFO: load addon information from /afs/cern.ch/work/a/agridin/private/HELAC-Onia-2.5.5/addon/addon_process.dat
INFO: load initial variables from /afs/cern.ch/work/a/agridin/private/HELAC-Onia-2.5.5/input/default.inp
INFO: update initial variables from /afs/cern.ch/work/a/agridin/private/HELAC-Onia-2.5.5/input/user.inp
HO>generate g g > cc~(3S11) g
HO>
```

2) But, it is better to use Pythia8 interface (see $Pythia8/examples/main35.cc). One needs to provide the path to HELAC-Onia.

At SPDroot:
Since it is a part of Pythia8Plugins (not included in FairSoft) it wasn't possible to use the Pythia8->HO interface from the SPDroot.

But, Pythia8 can read LHE files. **Works well on lxpub.jinr.ru**

Les Houches Events (LHE) format

gg > J/psi g

```
</header>
<init>
    2212    2212  1.350000E+01  1.350000E+01       0       0   10042   10042       3       1
    3.0582739109E+01    4.1883769210E-02    1.0000000000E+00    81
</init>
<event>
     4    81  1.963075E-03  3.688693E+00  7.299270E-03  2.652635E-01
      21   -1    0    0  104  101  0.0000000000E+00  0.0000000000E+00  2.9636152414E+00  2.9636152414E+00  0.0000000000E+00
  0.000000E+00  9.0000E+00
      21   -1    0    0  101  102  0.0000000000E+00  0.0000000000E+00 -2.7352075285E+00  2.7352075285E+00  0.0000000000E+00
  0.000000E+00  9.0000E+00
     443    1    1    2    0    0  1.9101886410E+00  6.1026309432E-01  1.7332042825E-01  3.6927628761E+00  3.0960000000E+00
  0.000000E+00  9.0000E+00
      21    1    1    2  104  102 -1.9101886410E+00 -6.1026309432E-01  5.5087284647E-02  2.0060598938E+00  0.0000000000E+00
  0.000000E+00  9.0000E+00
  <rwgt>
    <wgt id='1001'> 0.19631E-02 </wgt>
    <wgt id='1002'> 0.12395E-02 </wgt>
    <wgt id='1003'> 0.33449E-02 </wgt>
    <wgt id='1004'> 0.10549E-02 </wgt>
    <wgt id='1005'> 0.66603E-03 </wgt>
    <wgt id='1006'> 0.17974E-02 </wgt>
    <wgt id='1007'> 0.45408E-02 </wgt>
    <wgt id='1008'> 0.28670E-02 </wgt>
    <wgt id='1009'> 0.77370E-02 </wgt>
  </rwgt>
</event>
```

Beam E

PDFs

The generated events are stored in helaconiarun/PROC_HO_0/P0_calc_0/output

HELAC-Onia addons

With the fitted parameters, we wrote a simple event generator for $pp(\bar{p}) \to \mathcal{Q} + X$, where $\mathcal{Q} = J/\psi, \psi(2S), \Upsilon(1S), \Upsilon(2S), \Upsilon(3S), \chi_{c0}, \chi_{c1}, \chi_{c2}$ and $\chi_{bJ}(nP)$ with $J = 0, 1, 2$,$n = 1, 2, 3$. The code is put in the subdirectory **addon/pp_psiX_CrystalBall**. One can drive such program with the following command line

```
HO> generate addon 2
```

Some special input parameters can be specified in **pp_psiX_CrystalBall/input**. One can set the type of $\mathcal{Q}$ in **state.inp** and its polarization in **polarization.inp**. The file **crystalball.inp** is used to input the parameters λ,κ,n and $\langle P_T \rangle$. We have performed some applications in Ref. [110];

A set of examples for $pp \to \psi(1S,2S) + X$, $pp \to \Upsilon(1S,2S,3S) + X$, $pp \to J/\psi J/\psi$ (and many more) is available.