Manual for the JHU generator

For simulation of a single-produced resonance at hadron colliders
(version v6.9.8, release date Apr. 11, 2016, manual date Apr. 11, 2016)

The generator from [1–3] is a model-independent generator for studying spin and parity properties of new resonances. Please cite [1–3] if using the "JHU generator". The code can be downloaded from [4]. The generator outputs LHE files which can be passed to parton shower programs for hadronization. Only relative values of cross sections are supposed to produce meaningful results, while absolute values are often subject to an arbitrary normalization.

Additionally, the package includes code for computing the matrix elements standalone which can be used in a numerical matrix element analysis. Please reference the above papers and refer to "MELA" when using the matrix element likelihood analysis technique. The latter was also introduced in Ref. [5]. The matrix element package (MELA) also depends on MCFM libraries for background parameterization which should be referenced [6] when used.

Contents

I. Installation 2

II. Configuration 2
A. Command line configuration 2
B. Configuration in parameter file 6
1. General parameters 7
2. Spin-0 parameters 7
3. Spin-1 parameters 9
4. Spin-2 parameters 9

III. Examples 10
A. $J^P = 0^+_m$ resonance, $X \to ZZ$ or WW 10
B. $J^P = 0^-_m$ resonance, $X \to ZZ$ or WW 10
C. $J^P = 0^+_m$ resonance, $X \to \gamma\gamma$ 10
D. $J^P = 0^-_m$ resonance, $X \to \gamma\gamma$ 11
E. $J^P = 2^+_m$ resonance, $X \to ZZ$ or WW or $\gamma\gamma$ 11
F. Cross-section calculation and fraction notation 11
G. LHAPDF 11

IV. JHU Generator Matrix Elements (JHUGenMELA) 11

V. Release notes 12

A. Specific configurations 16
1. "SM-like spin-zero", 0^+ 16
2. "Higher order spin-zero", 0^+_h 16
3. "Pseudoscalar spin-zero", 0^- 16
4. "Vector spin-one", 1^- 17
5. "Pseudovector spin-one", 1^+ 17
6. "Minimal Graviton, spin-two", 2^+ 17
7. "Higher order Graviton, spin-two", 2^+_h 17
8. "Higher order Graviton, spin-two", 2^-_h 18

References 19

I. INSTALLATION

Register and download the package from `www.pha.jhu.edu/spin` and untar the file. Go to the directory `JHUGenerator` where the code exists for generating events with the JHU Generator. In the `makefile`, you have two options for compiler, `Comp = ifort` or `Comp = gfort`. Then simply compile with:

```
$ make
```

II. CONFIGURATION

There are two ways to configure the program, from the command line and in the file `mod_Parameters.F90`. For documentation from the command line, one can use `JHUGen help`. In addition, the command line configurables are defined in the file `main.F90`. When one change the fortran code directly, one should also recompile the code for changes to take effect. In general, command-line configuration handles general event properties while the configuration file handles all of the couplings and physics handles.

A. Command line configuration

The list of command line configurables and the default values are (also defined in the `README`):

```
Collider:          1=LHC, 2=Tevatron, 0=e+e-
Process:           0=spin-0, 1=spin-1, 2=spin-2 resonance, 50=pp/ee->VH,
                   60=weakVBF, 61=pp->Hjj, 62=pp->Hj, 80=ttH, 90=bbH,
                   110=t+H t channel, 111=tbar+H t channel,
                   112=t+H s channel, 113=tbar+H s channel
MReso:             resonance mass in GeV (default=125.00)
GaReso:            resonance width in GeV (default=0.00407)
DecayMode1:        decay mode for vector boson 1 (Z/W/gamma)
DecayMode2:        decay mode for vector boson 2 (Z/W/gamma)
                     0=Z->2l,  1=Z->2q, 2=Z->2tau, 3=Z->2nu,
                     4=W->lnu, 5=W->2q, 6=W->taunu,
                     7=gamma, 8=Z->2l+2tau,
                     9=Z->anything, 10=W->lnu+taunu, 11=W->anything
PChannel:          0=g+g, 1=q+qb, 2=both
OffshellX:          Off-shellness option for resonance (X) in decay processes 0, 1 or 2
PDFSet:            1=CTEQ6L1(2001),  2=MSTW(2008),
                                     2xx=MSTW with eigenvector set xx=01..40,
                                     3=NNPDF3.0LO
VegasNc0:          number of evaluations for integrand scan
VegasNc1:          number of evaluations for accept-reject sampling
VegasNc2:          number of events for accept-reject sampling
Unweighted:        0=weighted events, 1=unweighted events
Interf:            0=neglect interference for 4f final states,
                   1=include interference
DataFile:          LHE output file
ReadLHE:           LHE input file from external file (only spin-0)
ConvertLHE:        Convert decay of the V from VH production.
                   Use DecayMode1 to specify the decay.
                   (should be a Z or W mode, depending on the input file)
TopDK:             For ttH or t+H, 0=leave top quarks as stable, 1=decay top quarks
FilterNLept:       For decay mode, reject events that have less than FilterNLept leptons
FilterOSPairs:     For decay mode, reject events that have less than FilterOSPairs pairs of
                     sign leptons of any flavor.
FilterOSSFPairs:   For decay mode, reject events that have less than FilterOSSFPairs pairs of
                     opposite-sign-same-flavor leptons.
CountTauAsAny:     For FilterOSSFPairs, taus can stand in place of electrons or muons
```

```
                    of the same charge.
LHAPDF:             PDF set to use if interfaced with LHAPDF. See below.
FacScheme:          PDF factorization scale scheme
MuFacMultiplier:    Multiplier for the factorization scale chosen by FacScheme
RenScheme:          QCD renormalization scale scheme
MuRenMultiplier:    Multiplier for the renormalization scale chosen by RenScheme
WidthScheme:        Higgs width scheme: 1 for running width, 2 for fixed width (default),
                    and 3 for the CPS
WidthSchemeIn:      For decay mode, reweight from one propagator to another by setting
                    WidthScheme and WidthSchemeIn to different values
ReweightDecay:      For decay mode, reweight input decay by the decay probability
PMZZEvals:          For ReweightDecay, number of evaluations per mass point (default: 200000)
WriteFailedEvents: Write events that fail in the LHE file, but with a weight of 0
                    (off by default)

Seed:               Random seed for event generation
pTjetcut:           Minimum pT for jets in GeV (default: 15)
deltaRcut:          Minimum deltaR for jets (default: 0.3)
mJJcut:             Minimum dijet mass in GeV (default: 0)
MPhotonCutoff:      Minimum mass for offshell photons in GeV, when included (default: 4)
(Couplings):        See below
```

A few more details on some particular parameters:

- `VegasNc0,1,2`: For unweighted event generation `VegasNc0` specifies the number of evaluations for the initial integrand scan. The actual event generation is controlled by either `VegasNc1` or `VegasNc2`. `VegasNc1` specifies the number of tries in the accept/reject phase and `VegasNc2` is the number of generated events. When generating unweighted events in `ReadLHE` mode, both `VegasNc1` or `VegasNc2` can be used to specify the number of generated events. For the generation of weighted events `VegasNc1` specifies the number of evaluations for each of 5 iterations during the initial integrand scan. `VegasNc2` gives the (approximate) number of generated weighted events.

- `OffshellX`: The flag specifies whether the events for processes 0, 1 or 2 feature a delta-function resonance mass (when set to 0) at or a finite-width distribution around the resonance mass (when set to 1). This option replaces the `OffXVV` numerical flag in previous versions.

- `PChannel`: This parameter is only meaningful in the spin-2 case and for the ttH and bbH production mechanisms. For spin-0, production is possible only via the gg process; for spin-1 and for the VBF and VH processes, production is only possible via the $q\bar{q}$ process. For $H + jj$ and $H + j$, all four parton channels gg, $q\bar{q}$, qg, and $\bar{q}g$ are included.

- `DecayMode1,2=7`: Valid for spin-0 and spin-2. Only OffXVV=000 or 100 are possible.

- In VH production and `ConvertLHE` mode, `DecayMode1` is used for the decay of the V. If it is a Z decay mode, ZH will be produced; if it is a W decay mode, WH will be produced.

- In ttH production, `DecayMode1` and `DecayMode2` are used to decay the W bosons from the top decay. Only W decay modes are allowed.

- In $t + H$ production, `DecayMode1` is used to decay the W boson from the top decay.

- `Interf`: For final states with 4 same flavor fermions, this parameter controls whether or not to include interference effects. It receives 0 or 1.

- `FacScheme`, `MuFacMultiplier`, and `RenScheme`, `MuRenMultiplier`: There are currently 10 different schemes, which set the basis of the scale up to the scale multiplier. A postive integer uses running scales per event whereas a negative one uses a fixed scale, and `MuFacMultiplier` and `MuRenMultiplier` determine the scale multipliers for the factorization and renormalization scales, respectively:

  - ± 0: $\mu_{F,\,R}$ are set to the default values of each process. The command line values of `MuFacMultiplier` and `MuRenMultiplier` are disregarded.

- ±1: $\mu_{F,R} \propto \sqrt{q_H^2}$ if the scheme number is positive, or $\mu_{F,R} \propto m_H$ if the scheme number is negative. $+1$ is the default value for `Process` 0, 1 and 2 with scale multiplier 0.5, and -1 is the default value for `Process` 50, 60, 61 and 62 with scale multiplier 1.
- ±2: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_{JJH}^2}$, where J refers to a particle (partons or leptons) immediately associated in the Higgs production. If the scheme number is negative and the Higgs and the associated partons originate from a common particle (e.g. V^* in VH), $\mu_{F,R} \propto m_{JJH}$.
- ±3: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_{JJ}^2} + \sqrt{q_H^2}$. If the scheme number is negative and the associated particles originate from a common particle without the Higgs (e.g. V in VH), $\mu_{F,R} \propto m_{JJ}+m_H$.
- ±4: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_{J1}^2}+\sqrt{q_{J2}^2}+\sqrt{q_H^2}$, where $J_{1,2}$ are the associated particles. If the scheme number is negative and the matrix element treats the associated particles as massive (e.g. t and $\bar{t}$ in $t\bar{t}H$), $\mu_{F,R} \propto m_{J1}+m_{J2}+m_H$. -4 is the default value for `Process` 80 and 90 with scale multiplier 0.5.
- ±5: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_{JJ}^2}$. If the scheme number is negative and the associated particles originate from a common particle without the Higgs (e.g. V in VH), $\mu_{F,R} \propto m_{JJ}$.
- ±6: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_{J1}^2} + \sqrt{q_{J2}^2}$, where $J_{1,2}$ are the associated particles. If the scheme number is negative and the matrix element treats the associated particles as massive (e.g. t and $\bar{t}$ in $t\bar{t}H$), $\mu_{F,R} \propto m_{J1} + m_{J2}$.
- ±7: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_{JH}^2}$, where J is the more massive associated particle by its pole mass. If the scheme number is negative and the matrix element treats an associated particle as massive (e.g. t in $t+H$), $\mu_{F,R} \propto m_J + m_H$ for the most massive associated particle.
- ±8: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_J^2}+\sqrt{q_H^2}$, where J is the more massive associated particle. If the scheme number is negative and the matrix element treats an associated particle as massive (e.g. t in $t+H$), $\mu_{F,R} \propto m_J + m_H$ for the most massive associated particle (same as option -7). -8 is the default value for `Process` 110, 111, 112 and 113. with scale multiplier 0.25.
- ±9: If the scheme number is positive, $\mu_{F,R} \propto \sqrt{q_J^2}$, where J is the more massive associated particle. If the scheme number is negative and the matrix element treats an associated particle as massive (e.g. t in $t+H$), $\mu_{F,R} \propto m_J$.

- `WidthScheme`: This option controls the width scheme in JHUGen. The options are 1 for running width, 2 for fixed width, and 3 for the complex pole scheme [7, 8]. 2 is the default option. Note that the parameter values have the same meaning as in POWHEG.

- `WidthSchemeIn`: In `ReadLHE` mode, if this is specified and is different than `WidthScheme`, the resonance propagator will be reweighted between the three options specified above by multiplying the LHE weight of each event by the ratio of the propagators.

- `ReweightDecay`: By default, in `ReadLHE` mode, output events are written with the same weight as in the input file. For a wide resonance, this is not fully correct. The full probability for a $gg \to X \to VV \to 4f$ event can be written as

$$P(gg \to X \to VV \to 4f) = P_{prod}(m_{4f}, \vec{\Omega}_{prod}) \times P_{dec}(m_{4f}) \times P_{dec}(\vec{\Omega}_{decay} \mid m_{4f})$$

  While the first factor is generated by the input generator and the third by JHUGen, by default the second factor, the decay contribution to the mass shape, is lost. If `ReweightDecay=1`, this term will be multiplied into the weight of the event written in the LHE file.

  - If POWHEG was generated with a fixed width line shape, the full $P_{dec}(m_{4f})$ is multiplied into the weight.
  - If POWHEG was generated with the CPS option, then $m_{4f}\Gamma_H$, which is proportional to $P_{X\to anything}(m_{4l})$, is already included in the line shape. This is not exactly correct, as what is needed is the more specific $P_{X\to(\text{decay mode})}$. The weight is therefore multiplied by the branching fraction.

  To select one of these options, POWHEG input decay mode is specified by `WidthSchemeIn`. Alternatively, JHUGen will try to read the parameter `bwshape` from the header of the POWHEG file. Using other input generators requires an advance knowledge of how the line shape is generated; if neither of these options applies it may be simpler to apply a similar weight after the fact rather than in the LHE file.

  - `PMZZEvals`: Controls the number of evaluations per mass point. The default is 200000 at high masses, corresponding to a precision of about 0.2%, but increases at lower `m_Reso`, where the offshell Z's make the integration less efficient.

- `WriteFailedEvents`: In `ReadLHE` mode, events that fail in decay or that fail the lepton filter are by default not written to the LHE file. If this option is set to 1, they are written, but their weight is set to 0. If it is set to 2, they are written with a weight of zero and with no particles in the event. This option is useful if the same number of events are required to be in the input and the output.

- `Seed`: To reproduce previous results, the random seed can be set on the command line. Because Fortran uses multiple random seeds (the exact number is compiler dependent), the seed provided is used along with other fixed seeds specified in `mod_Parameters.F90` to generate however many seeds are needed. If no seed is provided on the command line, the seed is determined from the system time. Either way, it is printed in the header of the output file so that the results can be reproduced.

  Please note that random number generation is compiler dependent, so to ensure reproducibility the same compiler (and preferably the same version of the compiler) should be used. To facilitate this, the compiler and version are written in the header of the LHE file. It is also not guaranteed that the results can be reproduced using a different version of the generator.

- `pTjetcut`, `deltaRcut`, `mJJcut` All three cuts are used in processes 60 (VBF), 61 ($H + jj$), 90 (bbH), and also 80 (ttH) if `m_Top` is set to be light (< 10GeV). In addition, `pTjetcut` is used in process 62 ($H + j$) and also processes 110–113 ($t/\bar{t} + H$) if `m_Top` is light.

- Couplings: anomalous couplings can be set on the command line. For the full list see sections II B 2, II B 3, and II B 4; the parameters can be set in the command line include:

  - `ghgX`, `ghzX`, `ghzX_primeY`, `cz_qXsq`, `ghwX`, `ghwX_primeY`, `cw_qXsq`, `ghzgsX`, `ghzgs1_prime2`, and `ghgsgsX` for spin 0 HVV couplings (see the list for the ranges of X and Y)
  - `zprime_qq_left`, `zprime_qq_right`, `zprime_zz_1`, and `zprime_zz_2` for spin 1
  - `aX`, `bX`, `graviton_qq_left`, `and graviton_qq_right` for spin 2 (see the list for the range of X)
  - The Hff couplings `kappa` and `kappa_tilde`

  The couplings are complex, and must be set with a comma between the real and imaginary parts. For example:

  ```
  ghz1=0,0 ghz4=1,0
  ```

  If an anomalous coupling is turned on, the default coupling in the same category must also be specified to leave it on or turn it off. For example, if `ghz4` is specified, `ghz1` must either be set to `0,0` to generate a pure pseudoscalar or to some other value to generate a CP-violating mixture.

  In the weak vector boson fusion process (VBF) we also allow for different ZZH and WWH couplings. Per default, they are assumed to be equal, set by the Z couplings; however, if any of the W couplings are set, they will be used instead for WWH fusion. WWH can be turned off entirely by explicitly setting one of them to `0,0`.

- `MPhotoncutoff` If the decay is to ZZ and any of the couplings to photons (`ghzgsX` or `ghgsgsX`) is set, the γ^* contribution will be included along with the Z's. Similarly, if the decay is to $Z\gamma$ and any of the `ghgsgsX` couplings are set, the $\gamma\gamma^*$ contribution will be included as well. In this case, a lower cut on the photon invariant mass has to be placed in order to avoid the collinear singularity. This can be set using MPhotonCutoff.

Examples of running the generator:

- gg production:

  ```
  ./JHUGen Collider=1 Process=0 VegasNc2=100000 PChannel=0 DecayMode1=0 \\
  DecayMode2=0 Unweighted=1 DataFile=ggHZZ4l
  ```

- gg production, pseudoscalar resonance:

  ```
  ./JHUGen Collider=1 Process=0 VegasNc2=100000 PChannel=0 DecayMode1=0 \\
  DecayMode2=0 Unweighted=1 DataFile=ggHZZ4l_0- ghz1=0,0 ghz4=1,0
  ```

- $ggH \to Z\gamma$

```
./JHUGen DecayMode1=0 DecayMode2=7 DataFile=ggHgammaZ2l ghzgs2=1
```

- ZH with hadronic Z decay (change `DecayMode1` for other Z decays; both pp and e^+e^- Collider options possible):

```
./JHUGen Collider=1 Process=50 DecayMode1=1 Unweighted=1 VegasNc2=100000 \\
OffXVV=011 DataFile=ZH
```

- WH with leptonic W decay (change `DecayMode1` for other W decays):

```
./JHUGen Collider=1 Process=50 DecayMode1=4 Unweighted=1 VegasNc2=100000 \\
OffXVV=011 DataFile=WH
```

- VBF:

```
./JHUGen Collider=1 Process=60 Unweighted=1 VegasNc2=100000 DataFile=VBF \\
pTjetcut=0 deltaRcut=0
```

  (Jet cuts are generally not needed for VBF production, since there is no divergence.)

- $H + jj$:

```
./JHUGen Collider=1 Process=61 Unweighted=1 VegasNc2=100000 DataFile=Hjj
```

- $H + j$:

```
./JHUGen Collider=1 Process=62 Unweighted=1 VegasNc2=100000 DataFile=Hj
```

- $pp \to t\bar{t} + H$ with inclusive top decay (change `DecayMode1,2` for specific tt decays):

```
./JHUGen Collider=1 Process=80 DecayMode1=11 DecayMode2=11 Unweighted=1 \\
VegasNc2=100000 DataFile=ttH
```

- $pp \to b\bar{b} + H$:

```
./JHUGen Collider=1 Process=90 Unweighted=1 VegasNc2=100000 DataFile=bbH
```

- $pp \to t + H$, t channel with inclusive top decay (change `DecayMode1` for specific t decays; similar syntax for processes 111–113):

```
./JHUGen Collider=1 Process=110 DecayMode1=11 Unweighted=1 \\
VegasNc2=100000 DataFile=tH_s
```

For generating Higgs decay in VBF, $H + j(j)$, VH, $ttbH$, or tH production modes by the JHU generator or NLO gluon fusion with another generator (e.g. POWHEG), use JHU generator in `ReadLHE` mode and specify the decay mode of interest (ZZ, WW, $\gamma\gamma$, $Z\gamma$), while the SM fermionic decays may be generated by Pythia without loss of generality.

B. Configuration in parameter file

In the file `mod_Parameters.F90`, one does all the configuration of the couplings of the resonance. After modifying this file, one needs to recompile.

1. General parameters

- Switches:
  - In the case when `PChannel=2` for a spin-2 resonance, the user can define an approximate ratio of the production of gg and $q\bar{q}$ production.

    ```
    fix_channels_ratio = .true.
    channels_ratio_fix = 0.25d0    ! desired ratio of N_qq/(N_qq+N_gg)
    ```

  - For VV decay (and similarly in ttH production), by default the V's are randomized so that, for example, `DecayMode1=5 DecayMode2=11` will produce all combinations that include at least one hadronic W decay (rather of specifically hadronic decay of the W^+). This can be switched off by changing

    ```
    logical, public, parameter :: RandomizeVV = .true.
    ```

    to `.false.`. In this case, `DecayMode1` will refer specifically to the W^+ and `DecayMode2` to the W^-. For ZZ decay, this parameter only determines whether the decays of first and second Z written to the LHE file are randomized or not; in this case the only effect is the appearance of the output.
  - When reading in LHE files, JHUGen determines the format by looking at the first lines of the first event. If this determination fails for any reason (for example, if different events are written in inconsistent formats), try changing

    ```
    +logical, public, parameter :: UseUnformattedRead = .false.
    ```

    to `.true.`. However, this will noticeably slow down event generation in `ReadLHE` mode.
  - For the generation of weighted events (command line `Unweighted=0`) an LHE output file is created if

    ```
    logical, public, parameter :: writeWeightedLHE = .false.
    ```

    is set to `.true.`.
- Jet cuts, as described above, which should generally be set by command line options.
- Constants, more or less self-explanatory, including the masses of particles and width of Standard Model particles and the generated resonance, CKM matrix elements, Z and W branching fractions and scale factors for NLO QCD corrections $(1+\alpha_s/\pi)$, and collider energies. They are documented in comments in `mod_Parameters.F90`.
- Resonance couplings to SM fields, described in detail in the next few sections. The couplings should be set using command line options.

2. Spin-0 parameters

The `*hg*` parameters control the coupling of a spin-0 resonance to gluons in the production mechanism. In practice, the production parameters do not have a large effect since angular corrections from the production mechanism are lost for spinless particles. The `*hz*` parameters control the decay into Z and W bosons. One has the options to set the spin-0 couplings either from Eq.(9) or Eq.(11) from Ref. [2]. To switch between the two, use the parameter `generate_as`. We allow for q^2 dependent form factors similar to those described in Ref. [3]:

$$
\begin{aligned}
g_i^{\Lambda}(q_1,q_2) &= g_i' \frac{\Lambda_i^4}{(\Lambda_i^2+|q_1^2|)(\Lambda_i^2+|q_2^2|)} \\
&+ g_i'^2 \frac{(q_1^2+q_2^2)}{\Lambda_i^2} + g_i'^3 \frac{(q_1^2-q_2^2)}{\Lambda_i^2} + g_i'^4 \frac{(q_1+q_2)^2}{\Lambda_Q^2} \\
&+ g_i'^5 \frac{((q_1^2)^2+(q_2^2)^2)}{\Lambda_i^4} + g_i'^6 \frac{((q_1^2)^2-(q_2^2)^2)}{\Lambda_i^4} + g_i'^7 \frac{q_1^2\,q_2^2}{\Lambda_i^4} \\
g_i(q_1,q_2) &= g_i\delta_{i1} + (g_i^{\Lambda}(q_1,q_2) + g_i(1-\delta_{i1})) \frac{\Lambda_{i1}^2\,\Lambda_{i2}^2\,\Lambda_{i3}^2}{(\Lambda_{i1}^2+c_{i1}|q_1|^2)\,(\Lambda_{i2}^2+c_{i2}|q_2|^2)\,(\Lambda_{i3}^2+c_{i3}|q_1+q_2|^2)}
\end{aligned}
$$

The user has the option to choose between these functional forms, where the term multiplying g_i' corresponds to the full functional form and the $g_i''...g_i''''''$ correspond to an expansion in Λ^2. All parameters can be modified in `mod_Parameters.F90` by:

```
!-- parameters that define on-shell spin 0 coupling to SM fields, see note
   logical, public, parameter :: generate_as = .false.
   complex(8), public, parameter :: ahg1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahg2 = (0.0d0,0d0)
   complex(8), public, parameter :: ahg3 = (0.0d0,0d0)  ! pseudoscalar
   complex(8), public, parameter :: ahz1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahz2 = (0.0d0,0d0)  ! this coupling does not contribute for
                                                        ! gamma+gamma final states
   complex(8), public, parameter :: ahz3 = (0.0d0,0d0)  ! pseudoscalar

!-- parameters that define off-shell spin 0 coupling to SM fields, see note
   complex(8), public, parameter :: ghg2 = (1.0d0,0d0)
   complex(8), public, parameter :: ghg3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghg4 = (0.0d0,0d0)   ! pseudoscalar
   complex(8), public, parameter :: ghz1 = (2.0d0,0d0)
   complex(8), public, parameter :: ghz2 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz4 = (0.0d0,0d0)   ! pseudoscalar

!-- parameters that define q^2 dependent form factors
   complex(8), public, parameter :: ghz1_prime = (0.0d0,0d0)
   complex(8), public, parameter :: ghz1_prime2= (0.0d0,0d0)
   complex(8), public, parameter :: ghz1_prime3= (0.0d0,0d0)
   complex(8), public, parameter :: ghz1_prime4= (0.0d0,0d0)
   complex(8), public, parameter :: ghz1_prime5= (0.0d0,0d0)
   complex(8), public, parameter :: ghz1_prime6= (0.0d0,0d0)
   complex(8), public, parameter :: ghz1_prime7= (0.0d0,0d0)

   complex(8), public, parameter :: ghz2_prime = (0.0d0,0d0)
   complex(8), public, parameter :: ghz2_prime2= (0.0d0,0d0)
   complex(8), public, parameter :: ghz2_prime3= (0.0d0,0d0)
   complex(8), public, parameter :: ghz2_prime4= (0.0d0,0d0)
   complex(8), public, parameter :: ghz2_prime5= (0.0d0,0d0)
   complex(8), public, parameter :: ghz2_prime6= (0.0d0,0d0)
   complex(8), public, parameter :: ghz2_prime7= (0.0d0,0d0)

   complex(8), public, parameter :: ghz3_prime = (0.0d0,0d0)
   complex(8), public, parameter :: ghz3_prime2= (0.0d0,0d0)
   complex(8), public, parameter :: ghz3_prime3= (0.0d0,0d0)
   complex(8), public, parameter :: ghz3_prime4= (0.0d0,0d0)
   complex(8), public, parameter :: ghz3_prime5= (0.0d0,0d0)
   complex(8), public, parameter :: ghz3_prime6= (0.0d0,0d0)
   complex(8), public, parameter :: ghz3_prime7= (0.0d0,0d0)

   complex(8), public, parameter :: ghz4_prime = (0.0d0,0d0)
   complex(8), public, parameter :: ghz4_prime2= (0.0d0,0d0)
   complex(8), public, parameter :: ghz4_prime3= (0.0d0,0d0)
   complex(8), public, parameter :: ghz4_prime4= (0.0d0,0d0)
   complex(8), public, parameter :: ghz4_prime5= (0.0d0,0d0)
   complex(8), public, parameter :: ghz4_prime6= (0.0d0,0d0)
   complex(8), public, parameter :: ghz4_prime7= (0.0d0,0d0)

   real(8),    public, parameter :: Lambda_z1 = 10000d0*GeV
   real(8),    public, parameter :: Lambda_z2 = 10000d0*GeV
   real(8),    public, parameter :: Lambda_z3 = 10000d0*GeV
   real(8),    public, parameter :: Lambda_z4 = 10000d0*GeV
   real(8),    public, parameter :: Lambda_Q  = 10000d0*GeV
```

```
integer,    public, parameter :: cz_q1sq = 0d0 ! Sign of q1,2,12**2 for the following Lambda's
integer,    public, parameter :: cz_q2sq = 0d0
integer,    public, parameter :: cz_q12sq = 0d0

real(8),    public, parameter :: Lambda_z11 = 100d0*GeV ! For Z1
real(8),    public, parameter :: Lambda_z21 = 100d0*GeV
real(8),    public, parameter :: Lambda_z31 = 100d0*GeV
real(8),    public, parameter :: Lambda_z41 = 100d0*GeV
real(8),    public, parameter :: Lambda_z12 = 100d0*GeV ! For Z2
real(8),    public, parameter :: Lambda_z22 = 100d0*GeV
real(8),    public, parameter :: Lambda_z32 = 100d0*GeV
real(8),    public, parameter :: Lambda_z42 = 100d0*GeV
real(8),    public, parameter :: Lambda_z10 = 100d0*GeV ! For the Higgs
real(8),    public, parameter :: Lambda_z20 = 100d0*GeV
real(8),    public, parameter :: Lambda_z30 = 100d0*GeV
real(8),    public, parameter :: Lambda_z40 = 100d0*GeV
```

If the switch `includeGammaStar` is set to `.true.` then intermediate off-shell photons are included for Z boson final states. Their couplings to the spin-0 resonance are controlled by separate parameters,

```
complex(8), public, parameter :: ghzgs2  = (0.00d0,0d0)
complex(8), public, parameter :: ghzgs3  = (0.00d0,0d0)
complex(8), public, parameter :: ghzgs4  = (0.00d0,0d0)
complex(8), public, parameter :: ghgsgs2 = (0.00d0,0d0)
complex(8), public, parameter :: ghgsgs3 = (0.00d0,0d0)
complex(8), public, parameter :: ghgsgs4 = (0.00d0,0d0)
```

where the first three correspond to $Z\gamma^*$ couplings and the latter three corresponds to $\gamma^*\gamma^*$ interactions. These two sets of parameters also control the coupling strength in final states with on-shell photons, i.e. $Z\gamma$ and $\gamma\gamma$. The anomalous coupling involving the off-shell photon momentum (in γ^*Z interactions)

$$g_1' \frac{q_\gamma^2}{\Lambda_2^{Z\gamma}} m_Z^2 \epsilon_1^* \epsilon_2^*$$

is set by

```
complex(8), public, parameter :: ghzgs1_prime2= (0.0d0,0d0)
real(8),    public, parameter :: Lambda_zgs1 = 10000d0*GeV.
```

Each `ghzX`, `ghzX_primeY`, `Lambda_z`, and `cz_q1sq` parameter has a `ghwX`, `ghzwX_primeY`, `Lambda_w`, and `cw_q1sq` counterpart, which are used for the WWH coupling in VBF if at least one of them is set on the command line.

3. Spin-1 parameters

The parameters below represent the couplings given in Eq. (16) from Ref. [2]. The `*left*` and `*right*` parameters control the production of the spin-1 resonance while the `*_v` and `*_a` parameters control the decay.

```
!---parameters that define spin 1 coupling to SM fields, see note
   complex(8), public, parameter :: zprime_qq_left  = (1.0d0,0d0)   !  see note Eq. (4)
   complex(8), public, parameter :: zprime_qq_right = (0.0d0,0d0)
   complex(8), public, parameter :: zprime_zz_v =  (1.0d0,0d0)!  =1 for JP=1-
   complex(8), public, parameter :: zprime_zz_a =  (0.0d0,0d0)!  =1 for JP=1+
```

4. Spin-2 parameters

The `a*` parameters control the coupling of a spin-2 resonance to gluons in the production mechanism. The `b*` and `c*` parameters control the decay. One has the options to set the spin-2 couplings either from Eq.(18) or Eq.(19) from Ref. [2]. To switch between the two, use the parameter `generate_bis`.

```
  logical, public, parameter :: generate_bis = .true.
  logical, public, parameter :: use_dynamic_MG = .true. ! .true. (=default),
    ! the spin-2 resonance mass with MG^2=(p1+p2)^2, otherwise fixed at M_Reso^2.

  complex(8), public, parameter :: a1 = (1.0d0,0d0)    ! g1  -- c.f. note
  complex(8), public, parameter :: a2 = (0.0d0,0d0)    ! g2
  complex(8), public, parameter :: a3 = (0.0d0,0d0)    ! g3
  complex(8), public, parameter :: a4 = (0.0d0,0d0)    ! g4
  complex(8), public, parameter :: a5 = (0.0d0,0d0)    ! pseudoscalar, g8

  complex(8), public, parameter :: graviton_qq_left  = (1.0d0,0d0)! graviton coupling to quarks
  complex(8), public, parameter :: graviton_qq_right = (1.0d0,0d0)

  complex(8), public, parameter :: b1 = (1.0d0,0d0)
  complex(8), public, parameter :: b2 = (0.0d0,0d0)
  complex(8), public, parameter :: b3 = (0.0d0,0d0)
  complex(8), public, parameter :: b4 = (0.0d0,0d0)
  complex(8), public, parameter :: b5 = (0.0d0,0d0)
  complex(8), public, parameter :: b6 = (0.0d0,0d0)
  complex(8), public, parameter :: b7 = (0.0d0,0d0)
  complex(8), public, parameter :: b8 = (0.0d0,0d0)
  complex(8), public, parameter :: b9 = (0.0d0,0d0)
  complex(8), public, parameter :: b10 =(0.0d0,0d0)


  complex(8), public, parameter  :: c1 = (1.0d0,0d0)
  complex(8), public, parameter  :: c2 = (0.0d0,0d0)
  complex(8), public, parameter  :: c3 = (0.0d0,0d0)
  complex(8), public, parameter  :: c41= (0.0d0,0d0)
  complex(8), public, parameter  :: c42= (0.0d0,0d0)
  complex(8), public, parameter  :: c5 = (0.0d0,0d0)
  complex(8), public, parameter  :: c6 = (0.0d0,0d0)
  complex(8), public, parameter  :: c7 = (0.0d0,0d0)
```

III. EXAMPLES

The below examples are not meant to be a complete set, but rather some interesting and relevant cases. In many cases, the example is not the only way to produce such a scenario.

A. $J^P = 0^+_m$ resonance, $X \to ZZ$ or WW

```
./JHUGen ghz1=1,0 (...other options)
```

B. $J^P = 0^-_m$ resonance, $X \to ZZ$ or WW

```
./JHUGen ghz1=0,0 ghz4=1,0 (...other options)
```

C. $J^P = 0^+_m$ resonance, $X \to \gamma\gamma$

In practice, the example $X \to \gamma\gamma$ from this section, Sec. III C and the next Sec. III D are kinematically the same but are presented only to illustrate how one takes care of this final state.

```
./JHUGen DecayMode1=7 DecayMode2=7 OffXVV=100 ghgsgs2=1,0 (...other options)
```

D. $J^P = 0^-_m$ resonance, $X \to \gamma\gamma$

```
./JHUGen DecayMode1=7 DecayMode2=7 OffXVV=100 ghgsgs4=1,0 (...other options)
```

E. $J^P = 2^+_m$ resonance, $X \to ZZ$ or WW or $\gamma\gamma$

```
./JHUGen Process=2 a1=1,0 a2=0,0 b1=1,0 b2=0,0 b5=1,0 (...other options)
```

F. Cross-section calculation and fraction notation

For a vector boson coupling, we can represent the four independent parameters by two fractions (f_{g2} and f_{g4}) and two phases (ϕ_{g2} and ϕ_{g4}), defined for the HZZ and HWW couplings as follows (ignoring g_3)

$$f_{gi} = \frac{|g_i|^2\sigma_i}{|g_1|^2\sigma_1 + |g_2|^2\sigma_2 + |g_4|^2\sigma_4}\,; \qquad \phi_{gi} = \arg\left(\frac{g_i}{g_1}\right).$$

In order to obtain the cross-sections σ_i corresponding to the $g_i = 1$ coupling, generate large enough (e.g. VegasNc1=1000000, VegasNc2=50000000) number of weighted (`Unweighted=0`) with the corresponding couplings setup ($g_i = 1$, $g_{j\neq i} = 0$).

G. LHAPDF

It is possible to interface to an LHAPDF setup instead of compiling with local PDF's. To accomplish this:

- In the makefile:
  - Set UseLHAPDF=Yes
  - Set MyLHADir to a directory with your LHAPDF setup. This can be in terms of environment variables; for example `MyLHADir=${LHAPDF_DATA_PATH}/../../lib/`.
- Ensure that `$LHAPDF_DATA_PATH` and `$LD_LIBRARY_PATH` are set (both when compiling and when running).
- Compile
- Run with the extra command line parameter `LHAPDF` specifying your PDF set's `.info` file. For example: `LHAPDF=NNPDF30_lo_as_0130/NNPDF30_lo_as_0130.info`

IV. JHU GENERATOR MATRIX ELEMENTS (JHUGENMELA)

After extracting the code, you can go to the directory `JHUGenMELA` to find code for computing matrix elements directly. To compile the code, simply run `make`.

Please take note: The setup is configured for `gfort + gcc version 4.1.2 20080704 (Red Hat 4.1.2-50)` and it is highly dependent on the compiler version. Please configure for your own setup accordingly. (Using the `nm` command will help decipher the module names you will need)

Instructions for setting up the JHUGenMELA with MCFM are in the file `JHUGenMELA/ggZZ_MCFM/README`. This `C++` interface also covers the native matrix elements aforementioned through the wrapper class `TEvtProb`. The class currently includes the following calls:

- `TEvtProb`: Constructor with argument $\sqrt{s}$ in units of GeV.
- `SetProcess`: Set the process to be used in te calculation of the ME
- `SetMatrixElement`: Set the matrix element (Analytical, MCFM or JHUGen)
- `SetProduction`: Set the production (as enumerated in the TVar class)

- `SetLeptonInterf`: Set whether lepton interference is included in decay MEs. By default, it is included for JHUGen decay MEs or MCFM $q\bar{q}$ background MEs. It is not included in the other MCFM MEs.
- `ResetMCFM_EWKParameters`: Reset the MCFM EWK scheme and its parameters. See MCFM manual for the details on the EWK scheme.
- `Set_LHAgrid`: One-time call to setup the PDF path in case NNPDF 3.0 is used.
- `XsecCalc`: Calls to the decay MEs are handled through this function. Depending on the type of process, the self-defined coupling arrays may be ignored. Uses the following interfaced fortran subroutines:
  - "modhiggs__evalamp_gg_h_vv": Spin-0 matrix elements for gg initiated processes
  - "modzprime__evalamp_qqb_zprime_vv": Spin-1 matrix elements for $q\bar{q}$ initiated processes
  - "modzprime__evalamp_zprime_vv": Spin-1 matrix elements production-independent
  - "modgraviton__evalamp_qqb_g_vv": Spin-2 matrix elements for $q\bar{q}$ initiated processes
  - "modgraviton__evalamp_gg_g_vv:" Spin-2 matrix elements for gg initiated processes
  - "modgraviton__evalamp_g_vv": Spin-2 matrix elements production-independent
- `XsecCalcXJJ`: Calls to VBF, and Hjj MEs are handled from this function. Depending on the production mode or the process, these arrays may be ignored.
  - "modhiggsjj__evalamp_wbfh": Spin-0 matrix element for on-shell VBF production
  - "modhiggsjj__evalamp_sbfh": Spin-0 matrix element for on-shell $H + JJ$ QCD production
- `XsecCalcXJ`: Call to the JHUGen Hj ME is handled through tis routine.
  - "modhiggsj__evalamp_hj": Spin-0 matrix element for on-shell $H + J$ QCD production
- `XsecCalc_VX`: Call to the JHUGen VH ME is handled from this routine.
  - "modvhiggs__evalamp_vhiggs": Spin-0 matrix element for on-shell VH production
- `XsecCalc_TTX`: Call to te $t\bar{t}H$ ME is handled through this routine.
  - "modttbhiggs__evalxsec_pp_ttbh": Spin-0 matrix element for on-shell $t\bar{t}H$ production
  - "modttbhiggs__evalxsec_pp_bbbh": Spin-0 matrix element for on-shell $b\bar{b}H$ production
- `SetHiggsMass`: This routine changes the Higgs mass and width used in the MCFM calculations.

The inputs are usually the 4-vectors of the incoming patrons and outgoing particles in the CM frame of the object X. In addition the mass and width of the resonance are required as well as the ID of the outgoing particles. Finally the last set of inputs are the couplings themselves. They are arrays for parameters for a given spin hypothesis which mirror the parameters configurable in `mod_Parameters.F90`. Exemplary array initialization can be found in `testprogram.c` for the native matrix elements or `testME.C` for the interface with MCFM. For the interface, you can also see the beginning declarations in `TVar.hh` for the list of enumerated variables, and `testME.C` for example calls (eg. array filling, assignment of momenta, and passing te arguments to these routines).

V. RELEASE NOTES

In going from `v6.9.5` to `v6.9.8`, the updates are as follows:

- Standalone XVV production with V decay to any final state is improved.

In going from `v6.8.4` to `v6.9.5`, the updates are as follows:

- $X \to Z\gamma$ for spin-2 process `2` is implemented.
- Improvements are added to the cross section and phasespace in standalone productions for processes `0-2`.
- In `ReadLHE` mode, the mass shape can be reweighted:

  - to change from one propagator scheme to another, and/or
  - to account for $P_{dec}(m_{4f})$
- Additional command line parameters have been added:
  - Resonance width
  - Anomalous XVV and Xff couplings
  - Jet cuts
  - Random seed
- The command line option `OffXVV` (000-111) is replaced with the option `OffshellX` (`1/0`, `1` as default).

In going from `v6.7.8` to `v6.8.4`, the updates are as follows:

- Added running renormalization scale and generalized the running scales
- Added more options for the Higgs mass shape
- Improved LHE reading and writing

In going from `v6.2.8` to `v6.7.8`, the updates are as follows:

- Added running factorization scale ($\mu_F = m_{H*}/2$) in spin-0,1,2 decays
- Enabled identical 4ν and $4q$ final state interference in ZZ decay
- Faster VBF and Hjj event generation with deterministic jet flavor assignment
- VH code clean-up
- Update of the CKM matrix and of hadronic W decay
- New processes `110-113` for $t/\bar{t} + H$ production
- More flexible lepton filter

In going from `v5.6.3` to `v6.2.8`, the updates are as follows:

- Flexible LHE reading process for compatibility with more generators
- Small bugfixes

In going from `v5.2.5` to `v5.6.3`, the updates are as follows:

- Add `Process=90` for bbH production
- Add lepton filtering option
- Allow W from ttH to decay to any decay mode
- Allow W to decay to off-diagonal elements of the CKM matrix
- Add support for LHAPDF linking
- Fixes for LHE printout in VBF, Hjj, and VH

In going from `v4.8.1` to `v5.2.5`, the updates are as follows:

- Add `Process=80` for ttH production, with optional top decays
- Add support for NNPDF
- Make `DecayMode1` $\neq$ `DecayMode2` equivalent to generating everything and then filtering
- Add option for randomizing the V's in HVV decays

- Fixes for smoother reading of LHE files: mother assignment and invariant mass for all intermediate particles
- Add `ConvertLHE` option for converting VH decay to any DecayMode
- In `ReadLHE` and `ConvertLHE`, preserve comments and optional tags from the input LHE

In going from `v4.5.2` to `v4.8.1`, the updates are as follows:

- More flexibility for q^2-dependent form factors
- Separate couplings for ZZH and WWH in weak boson fusion
- Add new process: $pp \to H$+jet (Process=62)
- Extended LHE output format to allow for more digits
- MCFM plug-in for anomalous couplings in off-shell Higgs boson production in gg-¿ZZ
- Synchronize JHUGenMELA with MCFM library v6.8
- JHUGenMELA: extended MCFM ggHZZ matrix elements by anomalous couplings
- JHUGenMELA: add matrix elements for H+jet and V+H

In going from `v4.3.2` to `v4.5.2`, the updates are as follows:

- Add an option of intermediate photons for the modes with Z-bosons
- More flexibility for q^2-dependent form factors
- Option of hadronic branching rescaling (NLO QCD corrections) for inclusive decays
- Synchronize JHUGenMELA with the generator and with MCFM library v6.7

In going from `v4.2.1` to `v4.3.2`, the updates are as follows:

- Update LHE file format and index of partons
- Improve log printout
- Update ReadLHE mode: $H \to Z\gamma$ output and more flexible input
- VH production (replaces beta version)
- More flexibility for q^2-dependent form factors
- Tune q^2-dependence of couplings for some of the spin-2_h models
- Synchronize JHUGenMELA with the generator

In going from `v4.0.x` to `v4.2.x`, the updates are as follows:
To JHUGenerator:

- Fix BR in "all" decay mode
- Updates to LHE output
- Option to print out CS_max, output for g' and Lambdas
- Introduction of AnalyticMELA for $ee \to ZH$ and $pp \to ZH$ and analytic parton distribution functions

In going from `v3.1.x` to `v4.0.x`, the updates are as follows:
To JHUGenerator:

- Addition of VBF and Hjj process channels
- Possibility to read in VBF LHE event files

To JHUGenMELA:

- Interface with the MCFM program for ggZZ process
- Matrix elements for VBF and Hjj processes

In going from `v2.2.6` to `v3.1.8`, the updates are as follows:
To `JHUGenerator`:

- Capability reading LHE files with Higgs boson production, allows NLO production of spin-0;
- Extended the list of final state combinations;
- Log messages, lhe file headers, and minor cleanup.
- Updates to deal with non-zero lepton masses, lhe file format, and adjust default settings (e.g. lepton interference applied by default and can be configured in command line)

To `JHUGenMELA`:

- Production-independent JHUGenMELA for spin-0, 1, 2;
- Complex couplings in JHUGenMELA input.

In going from `v2.2.3` to `v2.2.6`, the updates are as follows:

- A small fix which corrects the *relative fraction* between the $2e2\mu$ and $4e/4\mu$ channels when using interference
- beta version is still under development
- $q\bar{q} \rightarrow$ spin-2 production is more safely performed with settings `PChannel = 2` and $q\bar{q}$ fraction = 1.

In going from `v2.1.3` to `v2.2.3`, the updates are as follows:

- Fix interference and randomization in the *beta* version
- Add the `JHUGenMELA` modules
- Small change for compilation on Mac OSX platforms
- Fix for tau masses in W decays

In going from `v2.0.2` to `v2.1.x`, the updates are as follows:

- Histograms are written in file (default: ./data/output.dat) and no longer on the screen. How to understand the histogram data and how to plot is briefly described in the output.dat file.
- Added tau masses
- Added lepton interference in the ZZ4l final state
- Added switch `generate_as` to choose couplings in spin-0 case (works for on- and off-shell resonance). The default is ".false.".
- Added the possibility to change graviton-quark couplings. The new parameters are `graviton_qq_left`, `graviton_qq_right` and correspond to $0.5*(1-\gamma^5)$ and $0.5*(1+\gamma^5)$ helicity projectors, respectively. Up to now the coupling was always vector-like. This is also the new default, `graviton_qq_left = graviton_qq_right =1`.
- The random seed is now fixed with gfortran.
- The call "./JHUGen help" prints out all available command line options
- Added new command line option "Unweighted=0 or 1" (default is 1)

APPENDIX A: SPECIFIC CONFIGURATIONS

We define configurations for certain models which are defined in Table 1 of [2].

1. "SM-like spin-zero", 0^+

```
!-- parameters that define on-shell spin 0 coupling to SM fields, see note
   logical, public, parameter :: generate_as = .false.
   complex(8), public, parameter :: ahg1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahg2 = (0.0d0,0d0)
   complex(8), public, parameter :: ahg3 = (0.0d0,0d0)  ! pseudoscalar
   complex(8), public, parameter :: ahz1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahz2 = (0.0d0,0d0)  ! this coupling does not contribute for gamma+gamma final states
   complex(8), public, parameter :: ahz3 = (0.0d0,0d0)  ! pseudoscalar

!-- parameters that define off-shell spin 0 coupling to SM fields, see note
   complex(8), public, parameter :: ghg2 = (1.0d0,0d0)
   complex(8), public, parameter :: ghg3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghg4 = (0.0d0,0d0)   ! pseudoscalar
   complex(8), public, parameter :: ghz1 = (1.0d0,0d0)
   complex(8), public, parameter :: ghz2 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz4 = (0.0d0,0d0)   ! pseudoscalar
```

2. "Higher order spin-zero", 0_h^+

```
!-- parameters that define on-shell spin 0 coupling to SM fields, see note
   logical, public, parameter :: generate_as = .false.
   complex(8), public, parameter :: ahg1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahg2 = (0.0d0,0d0)
   complex(8), public, parameter :: ahg3 = (0.0d0,0d0)  ! pseudoscalar
   complex(8), public, parameter :: ahz1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahz2 = (0.0d0,0d0)  ! this coupling does not contribute for gamma+gamma final states
   complex(8), public, parameter :: ahz3 = (0.0d0,0d0)  ! pseudoscalar

!-- parameters that define off-shell spin 0 coupling to SM fields, see note
   complex(8), public, parameter :: ghg2 = (1.0d0,0d0)
   complex(8), public, parameter :: ghg3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghg4 = (0.0d0,0d0)   ! pseudoscalar
   complex(8), public, parameter :: ghz1 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz2 = (1.0d0,0d0)
   complex(8), public, parameter :: ghz3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz4 = (0.0d0,0d0)   ! pseudoscalar
```

3. "Pseudoscalar spin-zero", 0^-

```
!-- parameters that define on-shell spin 0 coupling to SM fields, see note
   logical, public, parameter :: generate_as = .false.
   complex(8), public, parameter :: ahg1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahg2 = (0.0d0,0d0)
   complex(8), public, parameter :: ahg3 = (0.0d0,0d0)  ! pseudoscalar
   complex(8), public, parameter :: ahz1 = (1.0d0,0d0)
   complex(8), public, parameter :: ahz2 = (0.0d0,0d0)  ! this coupling does not contribute for gamma+gamma final states
   complex(8), public, parameter :: ahz3 = (0.0d0,0d0)  ! pseudoscalar

!-- parameters that define off-shell spin 0 coupling to SM fields, see note
   complex(8), public, parameter :: ghg2 = (0.0d0,0d0)
   complex(8), public, parameter :: ghg3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghg4 = (1.0d0,0d0)   ! pseudoscalar
   complex(8), public, parameter :: ghz1 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz2 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz3 = (0.0d0,0d0)
   complex(8), public, parameter :: ghz4 = (1.0d0,0d0)   ! pseudoscalar
```

4. "Vector spin-one", 1^-

```
!---parameters that define spin 1 coupling to SM fields, see note
   complex(8), public, parameter :: zprime_qq_left  = (1.0d0,0d0)
   complex(8), public, parameter :: zprime_qq_right = (0.0d0,0d0)
   complex(8), public, parameter :: zprime_zz_v =  (1.0d0,0d0)!  =1 for JP=1-
   complex(8), public, parameter :: zprime_zz_a =  (0.0d0,0d0)!  =1 for JP=1+
```

5. "Pseudovector spin-one", 1^+

```
!---parameters that define spin 1 coupling to SM fields, see note
   complex(8), public, parameter :: zprime_qq_left  = (1.0d0,0d0)
   complex(8), public, parameter :: zprime_qq_right = (0.0d0,0d0)
   complex(8), public, parameter :: zprime_zz_v =  (0.0d0,0d0)!  =1 for JP=1-
   complex(8), public, parameter :: zprime_zz_a =  (1.0d0,0d0)!  =1 for JP=1+
```

6. "Minimal Graviton, spin-two", 2^+

N.B. If an exclusive production mode is desired (e.g. $q\bar{q}$ or gg), this is handled at command-line configuration level via the `PChannel` variable.

```
!-- parameters that define spin 2 coupling to SM fields, see note
! minimal coupling corresponds to a1 = b1 = b5 = 1 everything else 0
  complex(8), public, parameter :: a1 = (1.0d0,0d0)    ! g1  -- c.f. draft
  complex(8), public, parameter :: a2 = (0.0d0,0d0)    ! g2
  complex(8), public, parameter :: a3 = (0.0d0,0d0)    ! g3
  complex(8), public, parameter :: a4 = (0.0d0,0d0)    ! g4
  complex(8), public, parameter :: a5 = (0.0d0,0d0)    ! pseudoscalar, g8
  complex(8), public, parameter :: graviton_qq_left  = (1.0d0,0d0)! graviton coupling to quarks
  complex(8), public, parameter :: graviton_qq_right = (1.0d0,0d0)

!-- see mod_Graviton
  logical, public, parameter :: generate_bis = .true.
  logical, public, parameter :: use_dynamic_MG = .true.

  complex(8), public, parameter :: b1 = (1.0d0,0d0)    !  all b' below are g's in the draft
  complex(8), public, parameter :: b2 = (0.0d0,0d0)
  complex(8), public, parameter :: b3 = (0.0d0,0d0)
  complex(8), public, parameter :: b4 = (0.0d0,0d0)
  complex(8), public, parameter :: b5 = (1.0d0,0d0)
  complex(8), public, parameter :: b6 = (0.0d0,0d0)
  complex(8), public, parameter :: b7 = (0.0d0,0d0)
  complex(8), public, parameter :: b8 = (0.0d0,0d0)
  complex(8), public, parameter :: b9 = (0.0d0,0d0)
  complex(8), public, parameter :: b10 =(0.0d0,0d0)  ! this coupling does not contribute for gamma+gamma final states


  complex(8), public, parameter  :: c1 = (1.0d0,0d0)
  complex(8), public, parameter  :: c2 = (0.0d0,0d0)
  complex(8), public, parameter  :: c3 = (0.0d0,0d0)
  complex(8), public, parameter  :: c41= (0.0d0,0d0)
  complex(8), public, parameter  :: c42= (0.0d0,0d0)
  complex(8), public, parameter  :: c5 = (0.0d0,0d0)
  complex(8), public, parameter  :: c6 = (0.0d0,0d0)
  complex(8), public, parameter  :: c7 = (0.0d0,0d0)
```

7. "Higher order Graviton, spin-two", 2_h^+

```
!-- parameters that define spin 2 coupling to SM fields, see note
! minimal coupling corresponds to a1 = b1 = b5 = 1 everything else 0
  complex(8), public, parameter :: a1 = (0.0d0,0d0)    ! g1  -- c.f. draft
  complex(8), public, parameter :: a2 = (0.0d0,0d0)    ! g2
  complex(8), public, parameter :: a3 = (0.0d0,0d0)    ! g3
```

```
  complex(8), public, parameter :: a4 = (1.0d0,0d0)    ! g4
  complex(8), public, parameter :: a5 = (0.0d0,0d0)    ! pseudoscalar, g8
  complex(8), public, parameter :: graviton_qq_left  = (1.0d0,0d0)! graviton coupling to quarks
  complex(8), public, parameter :: graviton_qq_right = (1.0d0,0d0)

!-- see mod_Graviton
  logical, public, parameter :: generate_bis = .true.
  logical, public, parameter :: use_dynamic_MG = .true.

  complex(8), public, parameter :: b1 = (0.0d0,0d0)    !  all b' below are g's in the draft
  complex(8), public, parameter :: b2 = (0.0d0,0d0)
  complex(8), public, parameter :: b3 = (0.0d0,0d0)
  complex(8), public, parameter :: b4 = (1.0d0,0d0)
  complex(8), public, parameter :: b5 = (0.0d0,0d0)
  complex(8), public, parameter :: b6 = (0.0d0,0d0)
  complex(8), public, parameter :: b7 = (0.0d0,0d0)
  complex(8), public, parameter :: b8 = (0.0d0,0d0)
  complex(8), public, parameter :: b9 = (0.0d0,0d0)
  complex(8), public, parameter :: b10 =(0.0d0,0d0)  ! this coupling does not contribute for gamma+gamma final states


  complex(8), public, parameter  :: c1 = (1.0d0,0d0)
  complex(8), public, parameter  :: c2 = (0.0d0,0d0)
  complex(8), public, parameter  :: c3 = (0.0d0,0d0)
  complex(8), public, parameter  :: c41= (0.0d0,0d0)
  complex(8), public, parameter  :: c42= (0.0d0,0d0)
  complex(8), public, parameter  :: c5 = (0.0d0,0d0)
  complex(8), public, parameter  :: c6 = (0.0d0,0d0)
  complex(8), public, parameter  :: c7 = (0.0d0,0d0)
```

8. "Higher order Graviton, spin-two", 2_h^-

```
!-- parameters that define spin 2 coupling to SM fields, see note
! minimal coupling corresponds to a1 = b1 = b5 = 1 everything else 0
  complex(8), public, parameter :: a1 = (0.0d0,0d0)    ! g1  -- c.f. draft
  complex(8), public, parameter :: a2 = (0.0d0,0d0)    ! g2
  complex(8), public, parameter :: a3 = (0.0d0,0d0)    ! g3
  complex(8), public, parameter :: a4 = (0.0d0,0d0)    ! g4
  complex(8), public, parameter :: a5 = (1.0d0,0d0)    ! pseudoscalar, g8
  complex(8), public, parameter :: graviton_qq_left  = (1.0d0,0d0)! graviton coupling to quarks
  complex(8), public, parameter :: graviton_qq_right = (1.0d0,0d0)

!-- see mod_Graviton
  logical, public, parameter :: generate_bis = .true.
  logical, public, parameter :: use_dynamic_MG = .true.

  complex(8), public, parameter :: b1 = (0.0d0,0d0)    !  all b' below are g's in the draft
  complex(8), public, parameter :: b2 = (0.0d0,0d0)
  complex(8), public, parameter :: b3 = (0.0d0,0d0)
  complex(8), public, parameter :: b4 = (0.0d0,0d0)
  complex(8), public, parameter :: b5 = (0.0d0,0d0)
  complex(8), public, parameter :: b6 = (0.0d0,0d0)
  complex(8), public, parameter :: b7 = (0.0d0,0d0)
  complex(8), public, parameter :: b8 = (1.0d0,0d0)
  complex(8), public, parameter :: b9 = (0.0d0,0d0)
  complex(8), public, parameter :: b10 =(0.0d0,0d0)  ! this coupling does not contribute for gamma+gamma final states


  complex(8), public, parameter  :: c1 = (1.0d0,0d0)
  complex(8), public, parameter  :: c2 = (0.0d0,0d0)
  complex(8), public, parameter  :: c3 = (0.0d0,0d0)
  complex(8), public, parameter  :: c41= (0.0d0,0d0)
  complex(8), public, parameter  :: c42= (0.0d0,0d0)
  complex(8), public, parameter  :: c5 = (0.0d0,0d0)
  complex(8), public, parameter  :: c6 = (0.0d0,0d0)
  complex(8), public, parameter  :: c7 = (0.0d0,0d0)
```

[1] Y.Y. Gao, A. V. Gritsan, Z.J. Guo, K. Melnikov, M. Schulze and N. V. Tran, "Spin-Determination of Single-Produced Resonances at Hadron Colliders". Phys. Rev. D **81**, 075022 (2010). arXiv:1001.3396 [hep-ph].
[2] S. Bolognesi, Y.Y. Gao, A. V. Gritsan, K. Melnikov, M. Schulze, N. V. Tran and A. Whitbeck, "On the Spin and Parity of Single-Produced Resonance at the LHC". Phys. Rev. D **86**, 095031 (2012). arXiv:1208.4018 [hep-ph].
[3] I. Anderson, S. Bolognesi, F. Caola, Y.Y. Gao, A. V. Gritsan, C. B. Martin, K. Melnikov, M. Schulze, N. V. Tran, A. Whitbeck, Y. Zhou, "Constraining anomalous HVV interactions at proton and lepton colliders". Phys. Rev. D **89**, 035007 (2014). arXiv:1309.4819 [hep-ph].
[4] See webpage: www.pha.jhu.edu/spin
[5] S. Chatrchyan *et al.* [CMS Collaboration], "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC," Phys. Lett. B **716**, 30 (2012) [arXiv:1207.7235 [hep-ex]].
[6] J. M. Campbell and R. K. Ellis, "MCFM for the Tevatron and the LHC," Nucl. Phys. Proc. Suppl. **205-206**, 10 (2010) [arXiv:1007.3492 [hep-ph]]. See also http://mcfm.fnal.gov
[7] S. Goria, G. Passarino and D. Rosco, "The Higgs-boson lineshape," Nucl. Phys. B **864** (2012) [arXiv:1112.5517 [hep-ph]].
[8] G. Passarino, C. Sturm and S. Uccirati, "Higgs pseudo-observables, second riemann sheet and all that," Nucl. Phys. B **834** (2010) [arXiv:1001.3360 [hep-ph]].