CDF/DOC/MONTECARLO/PUBLIC/6333
Version 2.0
November 6, 2004

LesHouchesModule: A CDF-interface module for Les Houches Accords

S. Tsuno[a,1], A.S. Thompson[b], and E. Gerchtein[c]
for Simulation group

[a] *Department of Physics, Okayama University, Okayama, Japan*

[b] *Department of Physics and Astronomy, University of Glasgow, Glasgow, U.K.*

[c] *Department of Physics, Carnegie Mellon University, Pittsburgh, U.S.*

Abstract

The purpose of this module is to convert an event data produced by the ME-based event generators outside CDF offline code into our data format, so-called HEPG format, by using CDF-PYTHIA or CDF-HERWIG in CDF offline software. Since this module works as a part of CDF simulations (cdfGen and cdfSim), the input event data is directory passed through the detector simulations with proper run numbers. Functionally, STDHEP format or even unknown-format undefined in this module is handled. All ME-based event generators working inside CDF offline are also interfaced by this module. The LesHouchsModule performs as a junction between ME-based event generators and CDF simulations in AC++ framework, so that all parameters are controlled by the `tcl` file.

We expect that it could be not only useful to easily accommodate an input data file to CDF simulation without having a trouble using a showering and decaying MC outside CDF offline code, but also useful to easily handle the parameter settings via `tcl`. This is important especially for the large scale of MC production.

[1] Corresponding author, e-mail : tsuno@fnal.gov

1 Introduction

The approach of the Matrix Element (ME) calculation in Monte Carlo event generator has been increasing its importance. There are a large number of ME-based event generators, for example, Alpgen[1], CompHep[2], MadGraph[3], and Grappa[4] as well as the process specific event generators, VECBOS[5], MCFM[6], WGRAD[7], MC@NLO[8] and WbbGen[9], *etc*. To make events from those ME-based event generations reliable, their outputs are normally connected with the Parton Showering (PS)-generators of PYTHIA[10] or HERWIG[11]. In past years, the programming procedure for the connection between ME- and PS-generators completely depended on customizing each generator. Those complicated generation models has led to a big effort to implement them into the PS-generators (see `herprt` in VECBOS or WGAMMA[12], *etc* in RunI). In order to avoid those efforts, the common data format[13] with the common interface routines has been proposed at Les Houches Workshop on 2001[14]. That common accord has been available since the release of PYTHIA.6.2 and HERWIG.6.5.

The purpose of this module, named "`LesHouchesModule`", is to convert an event data produced by the ME-generators outside CDF offline code into our data format, so-called HEPG format, by using CDF-PYTHIA or CDF-HERWIG in CDF offline software. Since this module works as a part of CDF simulations (cdfGen and cdfSim), the input event data is directory passed through the detector simulations with proper run numbers. So that all parameters of PS-generators are controlled by the `tcl` file, users can easily use those parameters via `talk-to`. Also it will be easy to connect to the decay packages which treat the particle decay after getting an event, like TAUOLA[16] or QQ. As a further advantage, the spin information can be connected directly to those decay packages without any interfacing routines. Once the data file with a proper LHA format is given into `LesHouchesModule`, everything goes though inside CDF offline code. Since the events from the ME-generators can be produced by the standalone execution of ME-generators, it is potentially available to get event data from MCDB[17], Patriot[18], and SAM[19] project at Fermi lab. In addition, we optionally provide the theory-induced scheme for the parameter settings.

In case that the input event format from ME-generators is not suited by the format defined in `LesHouchesModule` due to some reasons(not defined, or different version of ME-generators *etc*), we prepare an instruction to add a new ME-generator in this module, or the replacement of a certain file will fix it. We also introduce the demonstration to make the "User-original event generator" using this LHA faculty. Note that although the existing-ME generators (MCFM and WGRAD) in CDF offline software is also interfaced by this module, they have a different path from this module. So that the `LesHouchesModule` does not provide any useful faculties. Users should follow thier parameter settings in `tcl` file.

While this module provides the LHA framework in CDF offline code for many ME-generators, we should mention some disadvantages. There is less scope for the studies of PS-generators such as fragmentation/hadronization model where users may need to modify the source code of PYTHIA/HERWIG, and no benefits from theorists by which the whole event samples are customized, for example as better matchings between ME and PS-evolutions of CKKW[20] or LL-subtraction[21]. In this case, the event data format after PS-generators will be STDHEP[22] format. The `LesHouchesModule` is also able to read this format, and skip the running of CDF-PYTHIA or CDF-HERWIG.

We expect that it could be not only useful to easily accommodate an input data file to CDF simulation without having a trouble using a showering and decaying MC outside CDF offline code, but also useful to easily handle the parameter settings via `tcl`. Thus, we emphasis the following benefits. At first, users does not need a knowledge for the treatment of the external ME-generators running within PYTHIA or HERWIG by standalone. Also, since the decay packages are already included in CDF offline software, users does not have a trouble to run it. Second, users can use the same parameter settings and same version of PS-generators provided by physics groups since all simulations except the input event data produced by ME-generators are used within the CDF offline framework. All parameters are controlled by `tcl` file so that users can trace back which parameters are used for the simulation. This feature is also useful for the systematic studies which are just changed by one bit in the `tcl` files. Third, since the input event file is directory passed through the detector simulations, the data size to keep in a hard disk will be minimum, and the file maintanance will be minimum effort. This is important especially for the large scale of MC production.

2 Descriptions

2.1 Program structure

The program structure is illustrated in Figure 1. We assume that the input event data is well-definded by LHA format and readable to the subroutines provided from ME-generators, or the data format may be STDHEP format. These settings are set via `talk-to`. The input file name is set in the `tcl` file, then users can select the PS-generators of PYTHIA or HERWIG in the case that the input data format is from ME-generator according to LHA format, while those PS-generators are inactivated in case of STDHEP format. All parameters of PS-generators are exactly the same with the built-in PYTHIA or HERWIG inside CDF offline code, and set in the `tcl` file, too. The parameter settings are interfaced from C++ world to Fortran world. The `LesHouchesModule` calls these PS-generators. The whole sequence in the data flow is controlled by `GenInputManager` and `GenOutputManager`, so that the primary records such as the run numbers in the header of the CDF data format are set in this modules. Thus, the input data is capable to process into the detector simulation directory.

2.2 `talk-to` parameters

The parameters required by the `LesHouchesModule` are so far:

LesGenType :

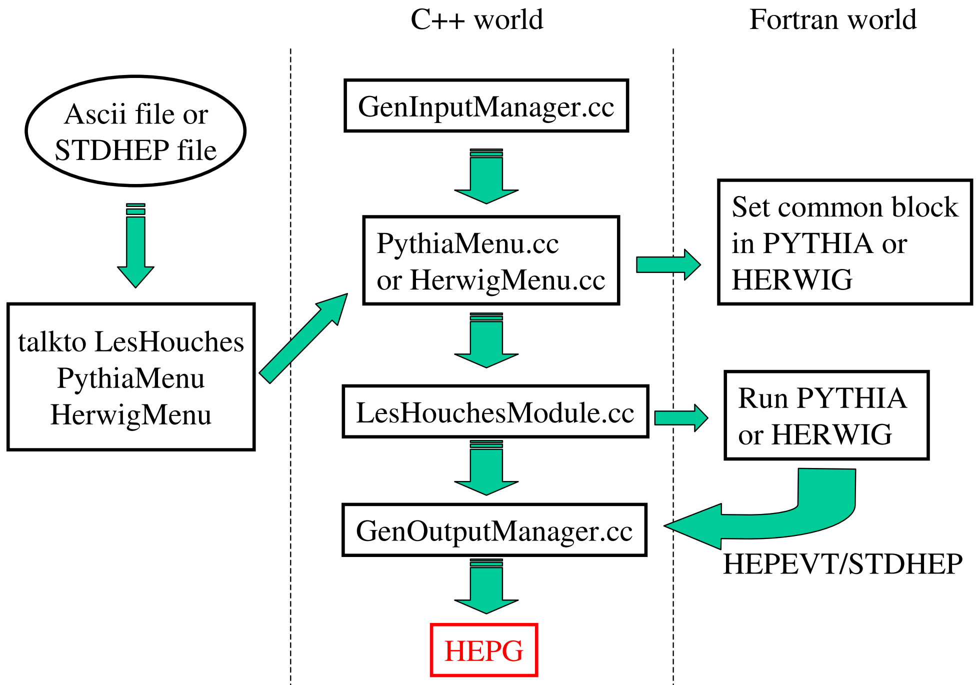

Figure 1: The data flow of "`LesHouchesModule`". Once the input file with the LHA format are given, those input events are internally converted to HEPG format by using the built-in PYTHIA or HERWIG modules in the CDF offline code.

specifies the name of ME-generators. The event format from those generators in the input file must be specified according to this unique ID name. Here note that the authors of each event generator DO NOT have a responsibility to decide this output event format itself. This is user's responsibility so that users should decide how to store the event record in the file. The event format thus strongly depends on those who use ME-generators. Users must care that even in the same event generator, the different ID name is used to read the event. Current allowed ID names are **AlpGen**, **CompHep**, **MadGraph**, **Grappa**, **Wgrad**, bf MCFM, **PYXTRA-MADGRAPH**, **MadGraph3**, and **USER**. If users meet a different format difinded in this module, the simple instruction will solve it. That is described in the later section.

LesDatFile :
specifies the input file name of the event file from the ME-generator. Users should check this event format is exactly same as that assumed in the interface set by **LesGenType**.

LesPSModel :
sets which model of **PYTHIA** or **HERWIG** is used.

firstEventToRead :

is the starting event number to store in HEPG. For example, `firstEventToRead set 1000` is that the first 1000 events in the input event file are skipped, and then store events in HEPG after 1000 events.

putResonance :
sets resonance particles (W or Z) in the event history, if lepton pair appears sequentialy in event. Note that this option is not allowed in HERWIG if the process has the double resonance production. (See manual of HERWIG 6.5.)

scaleOfKtClus :
sets energy scale for the parton shower based on kt-clustering algorithm[23]. In this case, the PS-scale is overwritten by a new scale recalculated by the kt-cluster. Otherwise, the original scale in the event file is used.

useTAUOLA :
is the option switch if TAUOLA is used. The helicity information of tau's is directory connected in TAUOLA if the input event file has the spin information of the final state particles.

useStdhep :
is the flag switch for STDHEP format. If `useStdhep set 1`, the events are stored as STDHEP file specified by **StdhepFileName** as well as HEPG. Note that the HEPG file is also produced at the same time. If `useStdhep set 2`, the events are read from STDHEP file specified by **StdhepFileName**, then stored in HEPG file. In this case, the input event file (**LesDatFile**) is ignored.

StdhepFileName :
specifies the output or input file name of the STDHEP file.

NumOfStdEvent :
is the event number to store/read in the STDHEP file set by **StdhepFileName**. For example, `NumOfStdEvent set 1000` is that the first 1000 events in the input event file of the ME-generator are recorded in the STDHEP file.

PythiaMenu :
is the "menu" for PYTHIA, which is the same as the `tcl` command in **PythiaModule**.

HerwigMenu :
is the "menu" for HERWIG, which is the same as the `tcl` command in **HerwigModule**.

As already mentioned, since the **LesHouchesModule** works as a part of CDF simulations of `cdfSim` and **cdfGen**, those parameters are also able to be set in those `tcl` files. The concrete example of the `tcl` file can be found in Appendix A.

2.3 Test the execution

Currently, this module contains six subroutines provided by each ME-generator: **ALPGEN**, **COMPHEP**, **MADGRAPH**, **GR@PPA**, **WGRAD**, and **MCFM**. We put the sample event data files for each generator in the directory `LesHouches/example/`. Users can test each event file and also check their event format (ASCII text) is the same manner with the format produced from thier ME-generator. However, the event format may be different depending on the program version of the ME-generator which users used. In this case, the simple instruction is given in the next section.

2.4 User interface

According to the LHA for a common data format with the common interface routines proposed at Les Houches Workshop on 2001, all ME-gerators must have two subroutines of **UPINIT** and **UPEVNT** if those correspond to the LHA. Those are all we need to modify for the user interface. In Figure 2, we illustrated the functions of these subroutines in **LesHouches** package within CDF offline framework. In order to manage several **UPINIT** and **UPEVNT** of ME-generators at the same time, the subroutine names are changed with respect to each name of the ME-generators (*ex*. `ALPGEN_UPINIT` *etc*). And since the input file is already opened at the initialization stage via `talk-to` to avoid the same I/O number, the lines related to the file-open in **UPINIT** are commented out. Thus, users may be able to handle their event file from the ME-generator just by replacing these subroutines with changing the name and comment out the file-open lines. The subroutines of **USER_UPINIT** and **USER_UPEVNT** with **USER** ID defined in **LesGenType** is dummy routines. Users may also use it. Users can also make an unique ME-generator ID is given by **LesGenType** in the `tcl` file. Look at **upinit.cc** and **upevnt.cc** in C++ world. It can be easly figured out how it works.

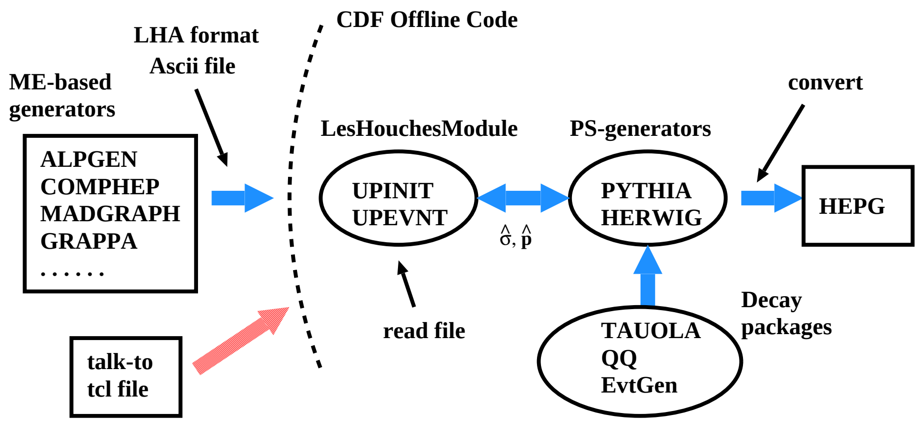

Figure 2: functions of **UPINIT** and **UPEVNT** in **LesHouches** package within CDF offline framework.

3 Basis to build up LHA model

In order to use the user original event generator, the LHA model can be followed to adopt any kind of generator to convert to the HEPG format in association with the parton showering by PYTHIA or HERWIG. In this section, we describe how to make the fake event generator that the emitted particles always have the same direction in the CDF detector. The detailed meaning of parameters are described in [13].

In the subroutine, `USER_UPINIT`, the initializations are required.

```
      SUBROUTINE USER_UPINIT
C...All real arithmetic in double precision.
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)

C...Include HEP common parameter.
      INTEGER MAXPUP
      PARAMETER (MAXPUP=100)
      INTEGER IDBMUP,PDFGUP,PDFSUP,IDWTUP,NPRUP,LPRUP
      DOUBLE PRECISION EBMUP,XSECUP,XERRUP,XMAXUP
      COMMON/HEPRUP/IDBMUP(2),EBMUP(2),PDFGUP(2),PDFSUP(2),
     &IDWTUP,NPRUP,XSECUP(MAXPUP),XERRUP(MAXPUP),XMAXUP(MAXPUP),
     &LPRUP(MAXPUP)
      SAVE /HEPRUP/

C...Set incoming beams: Tevatron Run II.
      IDBMUP(1) = 2212                ! proton
      IDBMUP(2) = -2212               ! anti-proton
      EBMUP(1) = 980D0                ! Beam energy
      EBMUP(2) = 980D0

C...Set PDF's of incoming beams: CTEQ 5L.
C...Note that Pythia will not look at PDFGUP and PDFSUP.
      PDFGUP(1) = 4                   ! PDF group
      PDFSUP(1) = 46                  ! PDF set
      PDFGUP(2) = PDFGUP(1)
      PDFSUP(2) = PDFSUP(1)

C...Decide on weighting strategy: unweighted on input.
      IDWTUP = 3

C...Number of external processes.
      NPRUP = 1                       ! Nr. of process
      LPRUP(1) = 200                  ! proc. ID
      XSECUP(1) = 1.0D0               ! cross section
      XMAXUP(1) = 1.0D0               ! max. weight

C...Done.
      RETURN
      END
```

The subroutine, `USER_UPEVNT`, fills up the particle information such as the 4-momentum, particle id's, energy scale, *etc.*

```
      SUBROUTINE USER_UPEVNT
C...All real arithmetic in double precision.
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)

C...Include HEP common parameter.
      INTEGER MAXNUP
      PARAMETER (MAXNUP=500)
      INTEGER NUP,IDPRUP,IDUP,ISTUP,MOTHUP,ICOLUP
      DOUBLE PRECISION XWGTUP,SCALUP,AQEDUP,AQCDUP,PUP,VTIMUP,SPINUP
```

```
      COMMON/HEPEUP/NUP,IDPRUP,XWGTUP,SCALUP,AQEDUP,AQCDUP,IDUP(MAXNUP),
     &ISTUP(MAXNUP),MOTHUP(2,MAXNUP),ICOLUP(2,MAXNUP),PUP(5,MAXNUP),
     &VTIMUP(MAXNUP),SPINUP(MAXNUP)
      SAVE /HEPEUP/

C...Flavour codes for entries.
      NUP = 4                       ! Nr. of particles
      IDUP(1) = 2                   ! u
      IDUP(2) = -2                  ! ubar
      IDUP(3) = 22                  ! photon

C...One example: unphysical graviton!!
      IDUP(4) = 39                  ! graviton

C...Status codes for entries.
      ISTUP(1) = -1
      ISTUP(2) = -1
      ISTUP(3) = 1
      ISTUP(4) = 1

C...Position of the first and last mother particles for entries.
      MOTHUP(1,3) = 1
      MOTHUP(2,3) = 2
      MOTHUP(1,4) = 1
      MOTHUP(2,4) = 2

C...Possible color strings for entries.
      ICOLUP(1,1) = 501
      ICOLUP(2,2) = 501

C...4-momentum for entries.
      PBEAM = 100.0D0               ! Collision beam energy
      PHI   = ACOS(-1.0D0)/3.0D0    ! phi
      THETA = ACOS(-1.0D0)/4.0D0    ! theta
      AMAS3 = 0.0D0                 ! mass of particle 3
      AMAS4 = 100.0D0               ! mass of particle 4

      PUP(3,1) = PBEAM
      PUP(4,1) = PBEAM
      PUP(3,2) = -PBEAM
      PUP(4,2) = PBEAM

      E3 = (4.0D0*PBEAM**2+AMAS3**2-AMAS4**2)/4.0D0/PBEAM
      P3 = SQRT((E3+AMAS3)*(E3-AMAS3))
      PUP(1,3) = P3*SIN(THETA)*COS(PHI)
      PUP(2,3) = P3*SIN(THETA)*SIN(PHI)
      PUP(3,3) = P3*COS(THETA)
      PUP(4,3) = E3
      PUP(5,3) = AMAS3

      E4 = (4.0D0*PBEAM**2+AMAS4**2-AMAS3**2)/4.0D0/PBEAM
      PUP(1,4) = -PUP(1,3)
      PUP(2,4) = -PUP(2,3)
      PUP(3,4) = -PUP(3,3)
      PUP(4,4) = E4
      PUP(5,4) = AMAS4

C...Set Q^2 and couplings. (whatever...)
      SCALUP = 100.0D0
      AQEDUP = 1.0/127.0
      AQCDUP = 0.127D0
      IDPRUP = 200

C...Done.
      RETURN
      END
```

Nothing is needed in this example anymore, and there is no input event file, so that the null character must be set for the input file name.

```
LesGenType set "USER"
LesDatFile set ""
LesPSModel set "PYTHIA"
```

The output results are

```
                            Event listing (summary)

   I particle/jet KS     KF  orig    p_x      p_y      p_z       E        m

   1 !p+!         21    2212    0    0.000    0.000  980.000  980.000    0.938
   2 !pbar-!      21   -2212    0    0.000    0.000 -980.000  980.000    0.938
 ==============================================================================
   3 !u!          21       2    1   -0.645    0.217   99.999  100.001    0.000
   4 !ubar!       21      -2    2   -1.399    0.843  -99.999  100.012    0.000
   5 !u!          21       2    3   -0.645    0.217   99.999  100.001    0.000
   6 !ubar!       21      -2    4   -1.399    0.843  -99.999  100.012    0.000
   7 !gamma!      21      22    0   25.950   46.159   53.076   74.974    0.000
   8 !Graviton!   21      39    0  -27.994  -45.100  -53.076  125.039  100.000
 ==============================================================================
   9 gamma         1      22    7   25.950   46.159   53.076   74.974    0.000
  10 Graviton  X   1      39    8  -27.994  -45.100  -53.076  125.039  100.000
  11 ud_0      A   2    2101    1    0.645   -0.217  879.992  879.993    0.579
  12 uu_1bar   V   1   -2203    2    1.399   -0.843 -879.992  879.994    0.771
 ==============================================================================
                  sum: -1.00         0.000    0.000    0.000 1960.000 1960.000
```

How exotic!! This is one example. Users can make any kind of event passed though PYTHIA or HERWIG, then automatically convert to HEPG format. This faculty is very helpful for the study of toy Monte Carlos, or the detector efficiency study, as well as the study of the different algorithms of the hadronization of PYTHIA and HERWIG.

4 Example: Gluon fusion Higgs production, $gg \to H \to b\bar{b}$

4.1 Theoretical basis

To give more a realistic example, we consider the gluon fusion Higgs production process beginning by briefly summarizing theoretical basis for the production cross section.

The gluonic width of the SM Higgs boson at lowest order is expressed elsewhere. For example, according to [24], the production cross section is

$$\begin{aligned} \hat{\sigma}_{LO}(gg \to H) &= \sigma_0 M_H^2 \delta(\hat{s} - M_H^2) \\ \sigma_0 &= \frac{\pi^2}{8M_H^2}\Gamma_{LO}(gg \to H) = \frac{G_F \alpha_s^2}{288\sqrt{2}\pi}\left|\sum_Q A_Q^H(\tau_Q)\right|^2 , \end{aligned} \quad (1)$$

where the scaling variable is defined as $\tau_Q = 4M_Q^2/M_H^2$ (Q is a quark flavor). The form factor can be evaluated as

$$A_Q^H(\tau_Q) = \frac{3}{2}\tau_Q[1 + (1 - \tau_Q)f(\tau_Q)] \quad (2)$$

$$f(\tau_Q) \quad = \quad \begin{cases} \arcsin^2 \frac{1}{\sqrt{\tau_Q}} & \tau_Q \geq 1 \\ -\frac{1}{4}\left[\log \frac{1+\sqrt{1-\tau_Q}}{1-\sqrt{1-\tau_Q}} - i\pi\right]^2 & \tau_Q < 1 \end{cases} \quad .$$

Now, the Higgs has the finite width depending on the coupling strength, and then is forced to decay to the bottom quark pair. Hence the delta function ($\delta(\hat{s} - M_H^2)$) in Eq.(1) is replaced with

$$\delta(\hat{s} - M_H^2) \quad \Rightarrow \quad \frac{1}{\pi} \int \frac{d\hat{s}}{(\hat{s} - M_H^2)^2 + M_H^2 \Gamma_H^2(tot)} \quad \times \quad M_H \Gamma_H(H \to b\bar{b}) \quad . \tag{3}$$

By integrating over the whole phase space, we get the total cross section, $\sigma_{LO}(gg \to H \to b\bar{b})$,

$$\sigma_{LO}(gg \to H \to b\bar{b}) = \int \frac{d\tau}{\tau} dy d\cos\theta d\phi f_i^1(x_1, Q^2) f_j^2(x_2, Q^2) d\hat{\sigma}_{LO}(\hat{s}) \quad , \tag{4}$$

where,

$$\tau = x_1 x_2 \quad , \quad y = \frac{1}{2} \ln \frac{x_1}{x_2} \quad , \tag{5}$$

where $f_i^a(x_a, Q^2)$ is the parton distribution function (PDF) of the hadron a ($= p$ or $\bar{p}$), giving the probability to find a parton i with an energy fraction x_a at an probing virtuality of Q^2.

4.2 Results

The source code for the user routine is presented in Appendix B. Here, we would like to state a result of the cross section. For the comparison, we also run parallel with the same process in PYTHIA (`MSUB(102) = 1`). The physical parameters taken in this execution were exactly the same. The result was the following,

```
==============================================================================
I                                  I                            I            I
I            Subprocess            I      Number of points      I    Sigma   I
I                                  I                            I            I
I----------------------------------I----------------------------I    (mb)    I
I                                  I                            I            I
I N:o Type                         I    Generated         Tried I            I
I                                  I                            I            I
==============================================================================
I                                  I                            I            I
I   0 All included subprocesses    I       100000        292853 I  4.034E-10 I
I   4 User process 200             I        50298        148608 I  2.025E-10 I
I 102 g + g -> h0                  I        49702        144245 I  2.009E-10 I
I                                  I                            I            I
==============================================================================
```

This is the standard output provided, the last column shows the cross sections, the second line is for our process and the third one is for the built-in process of PYTHIA. We are not going to describe the generation parameters in detail, here. There is a good agreement within 0.8% of the cross sections. The slight difference is due to the "out-of-controlled" parameters by which the (partial) width of Higgs boson in PYTHIA is calculated by event-by-event.

5 Summary

We have developed a CDF-interface module for the Les Houches Accords, `LesHouchesModule`, which converts the input event to the HEPG format using the built-in PYTHIA or HERWIG interfaces inside the CDF offline code. It should be emphasized that it is not only easy to translated to PYTHIA or HERWIG via `tcl` files, but also easy to implement user-defined event generator.

6 Acknowledgements

This development has been carried out in a collaboration with many authors of the Monte Carlo event generator. The integration to CDF offline software has been carried out with MC simulation group of CDF experiment. We would like to thank all members. Particularly, we are grateful to H.J. Frisch for encouraging us and M.L. Mangano and S. Mrenna for fruitful discussions about the program coding.

References

[1] M.L. Mangano et al., Aplgen, a generator for hard multiparton processes in hadronic collisions; hep-ph/0206293.

[2] A. Pukhov et al., CompHEP - a package for evaluation of Feynman diagrams and integration over multi-particle phase space. User's manual for version 33.; hep-ph/9908288.

[3] F. Maltoni and T. Stelzer, MadEvent: Automatic event generation with MadGraph; hep-ph/0208156.

[4] S. Tsuno et al., *Comput. Phys. Commun.* **151** (2003) 216.

[5] F.A. Berends et al., *Nucl. Phys.* **B 357** (1991) 32.

[6] J. Campbell and R.K. Ellis, *Phys. Rev.* **D 65** (2002) 113007.

[7] U. Baur, S. Keller and D. Wackeroth, *Phys. Rev.* **D 59** (1999) 013992.

[8] S. Frixione, and B.R. Webber, *J. High Energy Phys.* **06** (2002) 029.

[9] M.L. Mangano et al., Multijet matrix elements and shower evolution in hadronic collisions: W b anti-b + n jets as a case study; hep-ph/0108069.

[10] T. Sjöstrand, *Comput. Phys. Commun.* **82** (1994) 74.

[11] G. Marchesini and B. R. Webber, *Nucl. Phys.* **B 310** (1988) 461;
G. Marchesini et al.,*Comput. Phys. Commun.* **67** (1992) 465.

[12] U. Baur, E.L. Berger, *Phys. Rev.* **D 41** (1990) 1476.

[13] E. Boos et al., Generic User Process Interface for Event Generators; hep-hp/0109068

[14] Physics at Tev Colliders Workshop in Les Houches, 2001.

[15] T. Sjöstrand et al., in "Z physics at LEP I", eds. G. Altarelli, et al., CERN 89-08 (Geneva, 1989), Vol. 3, p.327.

[16] S. Jadach et al., *Comput. Phys. Commun.* **76** (1993) 361
J. Kim et al., CDF Run II Interface to TAUOLA, a decay library for polarized τ leptons; CDF/DOC/MONTECARLO/PUBLIC/5719.

[17] H.J. Frisch, The merits of stand-alone Monte Carlo generation; CDF/PHYS/MONTECARLO/PUBLIC/5938.

[18] Patriot: Physics Analysis & Tools Required to investigate Our Theories; http://cepa.fnal.gov/patriot/

[19] SAMGrid Project; http://projects.fnal.gov/samgrid/atwork.html

[20] S. Catani et al., *J. High Energy Phys.* **11** (2001) 063.

[21] Y. Kurihara et al., *Nucl. Phys.* **B 654** (2003) 301.

[22] A common output format for Monte Carlo events (after PS-generators); http://cepa.fnal.gov/psm/stdhep/

[23] S. Catani et al., *Nucl. Phys.* **B 406** (1993) 187.

[24] R.K. Ellis et al., QCD and Collider Physics, Cambridge University Press.

A Example `tcl` file

```
# LesHouchesModule : CDF-Interface module for LesHouches accord.
#=================================================================
path enable AllPath
show path

# input module for generators
module input GenInputManager

mod enable LesHouchesModule
mod talk LesHouchesModule

#  LesGenType set "Alpgen"
#  LesDatFile set "test.data"

  LesGenType set "Grappa"
  LesDatFile set "grc_w3j_mw_mincut.data"

#  LesGenType set "CompHep"
#  LesDatFile set "comphep_test.dat"

#  LesGenType set "MadGraph"
#  LesDatFile set "unweighted_events.dat"

#  LesGenType set "User"
#  LesDatFile set ""

#  LesPSModel set "PYTHIA"
  LesPSModel set "HERWIG"

  show

#########################################################
# PYTHIA section
# ==============
# Most of all parameters are available in PYTHIA menu.
#########################################################
#!! only valid for PYTHIA was turned ON.
  PythiaMenu
    commonMenu

# choose PDF set:  (ex. CTEQ5L)
      set_mstp -index=41  -value=1
      set_mstp -index=42  -value=1
      set_mstp -index=51  -value=7
      set_mstp -index=52  -value=1
      set_mstp -index=58  -value=5

# ISR/FSR/Hadronization/multiple interaction ON
      set_mstp -index=61  -value=1
      set_mstp -index=71  -value=1
      set_mstp -index=111 -value=1

#########################################################
# Rick's parameters:
#        Set A: Double Gaussian with more initial-state radiation
#
#  Initial-State Radiation Max Scale Factor
#  Old ISR setting with more initial-state radiation
      set_parp -index=67  -value=4.0
#  Underlying Event Parameters
      set_mstp -index=81  -value=1
```

```
# Double Gaussian
      set_mstp -index=82  -value=4
# MPI Cut-Off
      set_parp -index=82  -value=2.0
#  Warm-Core: 50% of matter in radius 0.4
      set_parp -index=83  -value=0.5
      set_parp -index=84  -value=0.4
# Almost Nearest Neighbor
      set_parp -index=85  -value=0.9
      set_parp -index=86  -value=0.95
# Energy Dependence with E0=1.8 TeV (do not vary this!)
      set_parp -index=89  -value=1800.0
      set_parp -index=90  -value=0.25
#########################################################

    exit
  exit

#########################################################
# HERWIG section
# ==============
# Most of all parameters are available in HERWIG menu.
#########################################################
#!! only valid for HERWIG was turned ON.
  HerwigMenu

#!! DO NOT change this !!
    Process set 1705

    Randseed1 set 9999
    Randseed2 set 123457

    Masses
      top set 175
    exit

# default Clpow leads to no B-Baryons
    Hadronization
       Clpow set 1.26
    exit
  exit
  show
exit

talk DHOutput
  output create main_stream test.gen
  output path main_stream AllPath
  output enable main_stream
  output keepList main_stream LRIH_StorableBank EVCL_StorableBank HEPG_StorableBank
  output list
exit

begin -nev 20
show timer

exit
```

B Example code for gluon fusion Higgs production

```
C=======================================================================
C     CDF - Les Houches interface for User-based generator.
C
C     Date    02/19/03
C     Author  Soushi Tsuno
C
C     user_upinit    Initialization for UPINIT.
C     user_upevnt    Event generation for UPEVNT.
C=======================================================================
      SUBROUTINE USER_UPINIT
      IMPLICIT NONE

C...Include HEP common parameter.
      INTEGER MAXPUP
      PARAMETER (MAXPUP=100)
      INTEGER IDBMUP,PDFGUP,PDFSUP,IDWTUP,NPRUP,LPRUP
      DOUBLE PRECISION EBMUP,XSECUP,XERRUP,XMAXUP
      COMMON/HEPRUP/IDBMUP(2),EBMUP(2),PDFGUP(2),PDFSUP(2),
     &IDWTUP,NPRUP,XSECUP(MAXPUP),XERRUP(MAXPUP),XMAXUP(MAXPUP),
     &LPRUP(MAXPUP)
      SAVE /HEPRUP/

C...PDF common parameter.
      CHARACTER*20      PARM(20)
      DOUBLE PRECISION  VALUE(20)

C...Set incoming beams: Tevatron Run II.
      IDBMUP(1) = 2212          ! proton
      IDBMUP(2) = -2212         ! anti-proton
      EBMUP(1) = 980D0          ! Beam energy
      EBMUP(2) = 980D0

C...Initialization of PDFLIB.
      PARM(1) = 'Init0'
      VALUE(1) = 0.0d0
      CALL PDFSET(PARM, VALUE)

C...Set PDF's of incoming beams: CTEQ 4L.
      PARM(1) = 'Nptype'
      VALUE(1) = 1
      PARM(2) = 'Ngroup'
      VALUE(2) = 4
      PARM(3) = 'Nset'
      VALUE(3) = 46
      CALL PDFSET(PARM, VALUE)

C...Set PDF's of incoming beams: CTEQ 5L.
C...Note that Pythia will not look at PDFGUP and PDFSUP.
      PDFGUP(1) = VALUE(2)      ! PDF group
      PDFSUP(1) = VALUE(3)      ! PDF set
      PDFGUP(2) = PDFGUP(1)
      PDFSUP(2) = PDFSUP(1)

C...Decide on weighting strategy: unweighted on input.
      IDWTUP = 1

C...Number of external processes.
      NPRUP = 1                 ! Nr. of process
      LPRUP(1) = 200            ! proc. ID
      XMAXUP(1) = 9.0D-1        ! maximum weight

C...Done.
      RETURN
      END



C=======================================================================
      SUBROUTINE USER_UPEVNT
      IMPLICIT NONE

C...Include HEP common parameter.
      INTEGER MAXPUP
      PARAMETER (MAXPUP=100)
      INTEGER IDBMUP,PDFGUP,PDFSUP,IDWTUP,NPRUP,LPRUP
      DOUBLE PRECISION EBMUP,XSECUP,XERRUP,XMAXUP
      COMMON/HEPRUP/IDBMUP(2),EBMUP(2),PDFGUP(2),PDFSUP(2),
     &IDWTUP,NPRUP,XSECUP(MAXPUP),XERRUP(MAXPUP),XMAXUP(MAXPUP),
     &LPRUP(MAXPUP)
      SAVE /HEPRUP/

C...Include HEP common parameter.
      INTEGER MAXNUP
      PARAMETER (MAXNUP=500)
      INTEGER NUP,IDPRUP,IDUP,ISTUP,MOTHUP,ICOLUP
      DOUBLE PRECISION XWGTUP,SCALUP,AQEDUP,AQCDUP,PUP,VTIMUP,SPINUP
      COMMON/HEPEUP/NUP,IDPRUP,XWGTUP,SCALUP,AQEDUP,AQCDUP,IDUP(MAXNUP),
     &ISTUP(MAXNUP),MOTHUP(2,MAXNUP),ICOLUP(2,MAXNUP),PUP(5,MAXNUP),
     &VTIMUP(MAXNUP),SPINUP(MAXNUP)
      SAVE /HEPEUP/

      REAL*8 PI,GF,WCM,BETAF,Z1,Z2,PYR,ALPHAS2,PYMRUN
      REAL*8 SHAT,WHAT,AJACOB,X1,X2,BETA,GAMM,FACT,SUM
      REAL*8 AMH,AGH,MB,MT,MBRUN
      REAL*8 TAUMIN,TAUMAX,TAURE,GAMRE,THEMIN,THEMAX,THE,TAU
      REAL*8 YSTMIN,YSTMAX,YST
      REAL*8 COS3MIN,COS3MAX,COS3,PHI3
      REAL*8 UPV1,DNV1,USEA1,DSEA1,STR1,CHM1,BOT1,TOP1,GL1
      REAL*8 UPV2,DNV2,USEA2,DSEA2,STR2,CHM2,BOT2,TOP2,GL2
      REAL*8 TQ,RQ,LQ,PQ1,PQ2,AQ1,AQ2,AP,AQ
      REAL*8 SIGIN,BWIG,RDCOR,SIGOUT
      REAL*8 E3,P3,SIN3

C...beta function.
      BETAF(Z1,Z2)=SQRT(1.0D0-2.0D0*(Z1+Z2)+(Z1-Z2)**2)

C...Constant.
      PI = ACOS(-1.0D0)
      GF = 1.16639D-5

C...Set Mass and width
      AMH = 120.0D0             ! Higgs mass
      AGH = 0.00405             ! Width of Higgs
      MB  = 4.8D0               ! b-quark mass
      MT  = 175.0D0             ! top mass
      WCM = EBMUP(1) + EBMUP(2) ! tot. Beam energy

C...Phase space Jacobian.
      AJACOB = 1.0D0

C...Set kinematic variables.
      TAUMIN = 4.0D0*MB**2/WCM**2
      TAUMAX = 1.0D0
      TAURE = AMH**2/WCM**2
      GAMRE = AMH*AGH/WCM**2
      THEMIN = ATAN((TAUMIN - TAURE)/GAMRE)
      THEMAX = ATAN((TAUMAX - TAURE)/GAMRE)
      THE = THEMIN + (THEMAX - THEMIN)*PYR(0)
      TAU = GAMRE*TAN(THE) + TAURE
      AJACOB = AJACOB*(THEMAX - THEMIN)
     $     *((TAU - TAURE)**2 + GAMRE**2)/(GAMRE*TAU)

      YSTMIN = LOG(SQRT(TAU))
      YSTMAX = -YSTMIN
      YST = YSTMIN + (YSTMAX - YSTMIN)*PYR(0)
      AJACOB = AJACOB*(YSTMAX - YSTMIN)

      COS3MIN = -1.0D0
      COS3MAX = 1.0D0
      COS3 = COS3MIN + (COS3MAX - COS3MIN)*PYR(0)
      AJACOB = AJACOB*(COS3MAX - COS3MIN)

      PHI3 = PYR(0)*2.0D0*PI
      AJACOB = AJACOB*2.0D0*PI

C...Set x1,x2,s^,w^
      X1 = SQRT(TAU)*EXP(YST)
      X2 = SQRT(TAU)*EXP(-YST)
      SHAT = TAU*WCM**2
```

```
      WHAT = SQRT(TAU)*WCM

C...Set beta,gamma
      BETA = TANH(YST)
      GAMM = COSH(YST)
      FACT = 0.38937966D9/2.0D0/SHAT

C...External PDF
      CALL STRUCTM(X1,WHAT,
     $     UPV1,DNV1,USEA1,DSEA1,STR1,CHM1,BOT1,TOP1,GL1)
      CALL STRUCTM(X2,WHAT,
     $     UPV2,DNV2,USEA2,DSEA2,STR2,CHM2,BOT2,TOP2,GL2)
C...gluon from p, gluon from pbar
      SUM = GL1*GL2
      AJACOB = AJACOB*SUM

C...Phase space factor
      AJACOB = AJACOB*FACT/2.D0/PI**2*BETAF(MB**2/SHAT,MB**2/SHAT)

C...Calculate Gamma(gg -> H).
      TQ = 4.0D0*MT**2/SHAT
      IF (TQ.LT.1) THEN
         RQ = SQRT(1.0D0-TQ)
         LQ = LOG((1.0D0+RQ)/(1.0D0-RQ))
         PQ1 = -(LQ**2-PI**2)/4.0
         PQ2 = PI/2.0*LQ
         AQ1 = -TQ/2.0D0*(1.0D0+(1.0D0-TQ)*PQ1)
         AQ2 = -TQ/2.0D0*(1.0D0-TQ)*PQ2
         AQ = AQ1**2+AQ2**2
      ELSE
         AP = (ASIN(1.0D0/SQRT(TQ)))**2
         AQ = 9.0D0*(0.5D0*TQ*(1.0D0+(1.0D0-TQ)*AP))**2
      ENDIF

      SIGIN = GF/288.0D0/SQRT(2.0D0)/PI*ALPHAS2(WHAT)**2*SHAT**2*AQ

C...Boson Propagator
C       BWIG = WHAT/((SHAT-AMH**2)**2+AGH**2*AMH**2)
C...For s-dependence width.
      BWIG = WHAT/((SHAT-AMH**2)**2+AGH**2*SHAT)

C...Calculate Gamma(H -> bb).
      MBRUN = PYMRUN(5,SHAT)              ! MSbar b-mass.
      RDCOR = 1.0D0 + ALPHAS2(WHAT)/PI    ! Radiative correction
      SIGOUT = 3.0D0*GF/4.0D0/SQRT(2.0D0)/PI*MBRUN**2*WHAT*RDCOR

C...Differencial cross section
      XWGTUP = AJACOB*SIGIN*BWIG*SIGOUT

C...Flavour codes for entries.
      NUP = 4                   ! Nr. of particles
      IDUP(1) = 21              ! gluon
      IDUP(2) = 21              ! gluon
      IDUP(3) = 5               ! b
      IDUP(4) = -5              ! b

C...Status codes for entries.
      ISTUP(1) = -1
      ISTUP(2) = -1
      ISTUP(3) = 1
      ISTUP(4) = 1

C...Position of the first and last mother particles for entries.
      MOTHUP(1,3) = 1
      MOTHUP(2,3) = 2
      MOTHUP(1,4) = 1
      MOTHUP(2,4) = 2

C...Possible color strings for entries.
      ICOLUP(1,1) = 501
      ICOLUP(2,2) = 501
      ICOLUP(1,2) = 502
      ICOLUP(2,1) = 502
      ICOLUP(1,3) = 503
      ICOLUP(2,4) = 503
```

```
C...Particle 1
      PUP(1,1) = 0.0D0
      PUP(2,1) = 0.0D0
      PUP(3,1) = 0.5D0*X1*WCM
      PUP(4,1) = 0.5D0*X1*WCM
      PUP(5,1) = 0.0D0

C...Particle 2
      PUP(1,2) = 0.0D0
      PUP(2,2) = 0.0D0
      PUP(3,2) = -0.5D0*X2*WCM
      PUP(4,2) = 0.5D0*X2*WCM
      PUP(5,2) = 0.0D0

C...Particle 3
      E3 = WHAT/2.D0
      P3 = SQRT((E3-MB)*(E3+MB))
      SIN3 = SQRT((1.0D0 - COS3)*(1.0D0 + COS3))
      PUP(1,3) = P3*SIN3*COS(PHI3)
      PUP(2,3) = P3*SIN3*SIN(PHI3)
      PUP(3,3) = GAMM*(P3*COS3 + BETA*E3)
      PUP(4,3) = GAMM*(E3 + BETA*P3*COS3)
      PUP(5,3) = MB

C...Particle 4
      PUP(1,4) = -PUP(1,3)
      PUP(2,4) = -PUP(2,3)
      PUP(3,4) = GAMM*(-P3*COS3 + BETA*E3)
      PUP(4,4) = GAMM*(E3 - BETA*P3*COS3)
      PUP(5,4) = MB

C...Set Q^2 and couplings. (whatever...)
      SCALUP = WHAT
      AQEDUP = 1.0/127.0
      AQCDUP = ALPHAS2(WHAT)

C...Ensure proper normalization of weights for MODE=+-2
      IDPRUP = 200

C...Done.
      RETURN
      END
```