

R. Pittau

Granada, January/February 2008

# The ALPGEN Generator

## The ALPGEN Generator

M.L. Mangano, M. Moretti, F. Piccinini, R. P., A.D. Polosa, JHEP07(2003)001

- Collection of MC codes for many processes relevant in hadron–hadron collisions (TEVATRON and LHC)
- Exact LO matrix element calculations of  $d\hat{\sigma}$  based on the ALPHA algorithm
- Parton-level event generation (weighted and unweighted)
- Interface to Herwig/Pythia for the evolution of the partonic final state through parton shower with matching à la MLM
- ALPGEN can be downloaded from the code home page <http://mlm.home.cern.ch/mlm/alpgen/> where a detailed documentation is also available

## Up to now available processes

- $W + N$  jets,  $Z/\gamma + N$  jets,  $N \leq 6$
- $WQ\bar{Q} + N$  jets,  $Z/\gamma Q\bar{Q} + N$  jets ( $Q = b, t$ ),  $N \leq 4$
- $W + c + N$  jets,  $N \leq 5$
- $n W + m Z + p$  Higgs +  $l$  photons +  $N$  jets,  
 $n + m + p + l \leq 8$ ,  $N \leq 3$
- $Q\bar{Q} + N$  jets, ( $Q = b, t$ ),  $N \leq 6$
- $Q\bar{Q}Q'\bar{Q}' + N$  jets, ( $Q, Q' = b, t$ ),  $N \leq 4$
- $N$  jets,  $N \leq 6$
- $Q\bar{Q}H + N$  jets, ( $Q = b, t$ ),  $N \leq 4$
- $N$  photons +  $M$  jets,  $N > 0$ ,  $N + M \leq 8$  and  $M \leq 6$

## NEW processes

- single top production:  $tq, tb, tW, tbW$ , No extra jets
- $H + N$  jets,  $N \leq 3$ , effective  $ggH$  coupling
- $W +$  photons  $+ N$  jets, up to 2  $\gamma$ 's
- $W +$  photons  $+ Q\bar{Q} + N$  jets, up to 2  $\gamma$ 's
- $Q\bar{Q} + N$  photons  $+ N$  jets, ( $Q = b, t$ ),  $M + N \leq 6$

## Tutorial (How to use ALPGEN)

- Go to <http://mlm.home.cern.ch/mlm/alpgen/> and download in an empty directory the file `v210.tgz` by clicking:

**NEW:** The source code for version V2.10

- Then give the command:

```
> tar -xzvf v210.tgz
```

resulting in the following directory structure:

```
2Qlib      compare      Makefile     QQhlib      vbjetwork   wphqqlib    zqqwork
2Qphlib    compile.mk   Njetlib     QQhwork     wcjetlib    wphqqwork
2Qphwork   DOCS        Njetwork    toplib      wcjetwork   wqqlib
2Qwork     ft90V.tar.gz phjetlib    topwork     wjetlib     wqqwork
4Qlib      herlib      phjetwork   v210.tgz   wjetwork    zjetlib
4Qwork     hjetlib     prc.list    validation  wphjetlib   zjetwork
alplib     hjetwork    pylib       vbjetlib    wphjetwork  zqqlib
```

- It is possible to validate the whole package by giving the command:

> make validate

It will compile *all* processes and will run them with few points, comparing then the results with a library of results to see if the installation worked or whether there are differences.

Then change directory:

> cd validation

to see the result of the check:

> more val.summary

Let us now show how to run a specific process (e.g.  $Wb\bar{b} + 1 \text{ jet}$ ).

- Go to the corresponding directory:

```
> cd wqqwork
```

that looks like

```
alpgen.inc  cnfg.dat  input  Makefile  pdflnk  wqq.inc  wqqusr.f
```

- Then compile the code with the command

```
> make gen
```

that will generate the executable file

```
wqqgen
```

- Run it by giving the command

```
> wqqgen < input
```

(the default `input` file corresponds to  $Wb\bar{b} + 1 \text{ jet}$ )

- The result of the run is stored in a series of files contained in the directory, that now looks like

```
alpgen.inc  Makefile      wbbj.grid2  wbbj.stat   wqqgen      wqqusr.o
cnfg.dat    pdflnk         wbbj.mon   wbbj.top    wqq.inc
input      wbbj.grid1    wbbj.par   wbbj.wgt    wqqusr.f
```

Before discussing the output we show how to prepare

the input file



```
1          ! imode
wbbj      ! label for files
0 ! start with: 0=new grid, 1=previous warmup grid, 2=previous generation grid
10000 2    ! Nevents/iteration, N(warm-up iterations)
100000    ! Nevents generated after warm-up
*** The above 5 lines provide mandatory inputs for all processes
*** (Comment lines are introduced by the three asterisks)
*** The lines below modify existing defaults for the hard process under study
*** For a complete list of accessible parameters and their values,
*** input 'print 1' (to display on the screen) or 'print 2' to write to file
ihvy 5
njets 1
ickkw 0
ptjmin 20
ptbmin 20
etajmax 1
etabmax 1
drjmin 0.7
drbmin 0.7
```

- There are hidden parameters, with default values. To see the entire list type:

```
> ./wqqgen
```

- Then it appears on the screen:

```
Input RUN generation mode:
```

```
0: generate weighted events, no evt dumps to file
```

```
1: generate wgtd events, write to file for later unweighting
```

```
2: read events from file for unweighting or processing
```

```
or documentation modes:
```

```
3: print parameter options and defaults, then stop
```

```
4: write to par.list parameter options and defaults, then stop
```

```
5: write to prc.list complete list of processes, parameter  
options and defaults, scale choices, PDF, etc., then stop
```

- Now write

```
> 4
```

- to generate the file `par.list`, that reads:

-----

hard process code (not to be changed):

ihrd= 1

-----

Select pp (1) or ppbar (-1) collisions:

ih2= -1

-----

beam energy in CM frame (e.g. 7000 for LHC):

ebeam= 980.

-----

parton density set:

ndns= 5

NDNS	Set	Lambda_4	Lambda_5_2loop	Scheme
1	CTEQ4M	.298	.202	MS
2	CTEQ4L	.298	.202	MS
3	CTEQ4HJ	.298	.202	MS
4	CTEQ5M	.326	.226 (as=0.118)	MS
5	CTEQ5L *	.192	.144 (asLO=0.127)	MS
6	CTEQ5HJ	.326	.226 (as=0.118)	MS

7	CTEQ6M	.326	.226	(as=0.118)	MS
8	CTEQ6L	.326	.226	(as=0.118)	MS
9	CTEQ6L1	.215	.165	(asL0=0.130)	MS
10-50	CTEQ6xx	.326	.226	(as=0.118)	MS
101	MRST99 COR01	.321	.220		MS
102	MRST2001	.342	.239	(as=0.119 )	MS
103	MRST2001	.310	.214	(as=0.117 )	MS
104	MRST2001	.378	.267	(as=0.121 )	MS
105	MRST2001J	.378	.267	(as=0.121)	MS
106	MRST2002L0 *	.22	.167	(asL0=0.13)	MS

PDF sets followed by \* are obtained from a 1-loop analysis, and the relative values of Lambda refer to 1-loop.

The MSbar scheme is used by default with 1-loop structure functions.

In all cases the values of Lambda and loop order are set automatically by the code, The user only needs to input ndns

-----

scale option (process dependent):

iqopt= 1

Options for Factorization/renormalization scale Q:

iqopt=0 => Q=qfac

iqopt=1 =>  $Q=qfac*\sqrt{m_W^2+ \text{sum\_jets}(m_{tr}^2)}$

iqopt=2 => Q=qfac\*mW

iqopt=3 =>  $Q=qfac*\sqrt{m_W^2+ pt_W^2}$

iqopt=4 =>  $Q=qfac*\sqrt{\text{sum\_jets}(m_{tr}^2)}$

where:

-  $m_{tr}^2=m^2+pt^2$ , summed over heavy quarks and light jets

To select CKKW scale input 'ickkw 1'

(mandatory for later use of jet matching)

In imode=2 select clustering option 'cluopt':

cluopt=1: kperp propto pt(cluster) (default)

cluopt=2: kperp propto mt(cluster)

kperp is then rescaled by ktfac

-----

Q scale rescaling factor:

qfac= 1.

-----

CKKW scale option: set to 1 to enable jet-parton matching:

```
  ickkw= 0
  -----
  scale factor for ckkw alphas scale:
  ktfac=  1.
  -----
  number of light jets:
  njets= 0
  -----
  heavy flavour type for procs like WQQ, ZQQ, 2Q, etc(4=c, 5=b, 6=t):
  ihvy= 5
  -----
  charm mass:
  mc=  1.5
  -----
  bottom mass:
  mb=  4.7
  -----
  top mass:
  mt= 174.3
  -----
```

minimum pt for light jets:

ptjmin= 20.

-----

ptmin for bottom quarks (in procs with explicit b):

ptbmin= 20.

-----

ptmin for charm quarks (in procs with explicit c):

ptcmin= 20.

-----

minimum pt for charged leptons:

ptlmin= 0.

-----

minimum missing et:

metmin= 0.

-----

max|eta| for light jets:

etajmax= 2.5

-----

max|eta| for b quarks (in procs with explicit b):

etabmax= 2.5

```
-----
max|eta| for c quarks (in procs with explicit c):
etacmax= 2.5
-----
max abs(eta) for charged leptons:
etalmax= 10.
-----
min deltaR(j-j), deltaR(Q-j) [j=light jet, Q=c/b]:
drjmin= 0.699999988
-----
min deltaR(b-b) (procs with explicit b):
drbmin= 0.699999988
-----
min deltaR(c-c) (procs with explicit charm):
drcmin= 0.699999988
-----
min deltaR between charged lepton and light jets:
drlmin= 0.
-----
first random number seed (5-digit integer):
```



iseed1= 12345

-----

second random number seed (5-digit integer):

iseed2= 67890

-----

W decay modes, in imode=2:

iwdecmod= 1

-----

kt scale option. 1:kt propto pt, 2:kt propto mt:

cluopt= 1

-----

first random number seed for unweighting (5-digit integer):

iseed3= 12345

-----

second random number seed for unweighting (5-digit integer):

iseed4= 67890

- All these default values can be changed by writing in the input file, *in any place*, the changed values
- For example, in our input file, the default value `etajmax= 2.5` has been changed by the statement

```
etajmax 1
```

We are now ready to discuss

the output files

## wbbj.stat

W b bbar + 1 jets

W-> ell nu

=====

b mass: 4.7

Generation cuts for the partonic event sample:

Light jets:

ptmin= 20. |etamax|= 1. dR(j-j),dR(Q-j)> 0.7

b quarks:

ptmin= 20. |etamax|= 1. dR(QQ)> 0.7

Leptons:

ptmin(lep)= 0. |etamax|= 10. Et(miss)> 0. dR(l-j)> 0.

RUNNING PARAMETERS

-----

Electroweak parameters:

iewopt= 3:

input mW, mZ, GF calculate the rest

M(W)= 80.419 Gamma(W)= 2.04807653  
M(Z)= 91.188 Gamma(Z)= 2.44194427  
M(H)= 120. Gamma(H)= 0.  
gW= 0.65323291; sin<sup>2</sup>(thetaW)= 0.222246533; 1/aem(mZ)= 132.50698

Quark masses:

m(top)= 174.3 m(b)= 4.7

Beams' parameters:

beam1=proton, beam2=antiproton

Ebeam= 980. PDF set=CTEQ5L

as(MZ) [nloop= 1] = 0.127003172

starting generation of 10000. events

average ph-space eff= 0.342536138

avgwgt(pb)= 0.0152015651+- 0.00187552134 maxwgt= 7.65766369

unwgt eff = 0.00198514399

sub-processes:

jproc= 1 total(pb): 0.0108255596+- 0.0017552771  
jproc= 2 total(pb): 0.00236215767+- 0.000554153443  
jproc= 3 total(pb): 0.00201384775+- 0.000374062198

cumulated cross-section:

avgwgt(pb)= 0.0152015651+- 0.00187552134

starting generation of 10000. events

average ph-space eff= 0.436891083

avgwgt(pb)= 0.019895344+- 0.00268578981 maxwgt= 14.9961835

unwgt eff = 0.00132669383

sub-processes:

jproc= 1 total(pb): 0.0136988128+- 0.00249145625

jproc= 2 total(pb): 0.00288165339+- 0.000481786113

jproc= 3 total(pb): 0.00331487788+- 0.000890439387

cumulated cross-section:

avgwgt(pb)= 0.0167401609+- 0.00153770483

starting generation of 100000. events

average ph-space eff= 0.409294254

avgwgt(pb)= 0.017476239+- 0.000574471343 maxwgt= 26.1532933

unwgt eff = 0.000668223264

sub-processes:

jproc= 1 total(pb): 0.0114901936+- 0.000490463401

jproc= 2 total(pb): 0.00322185443+- 0.000225250821

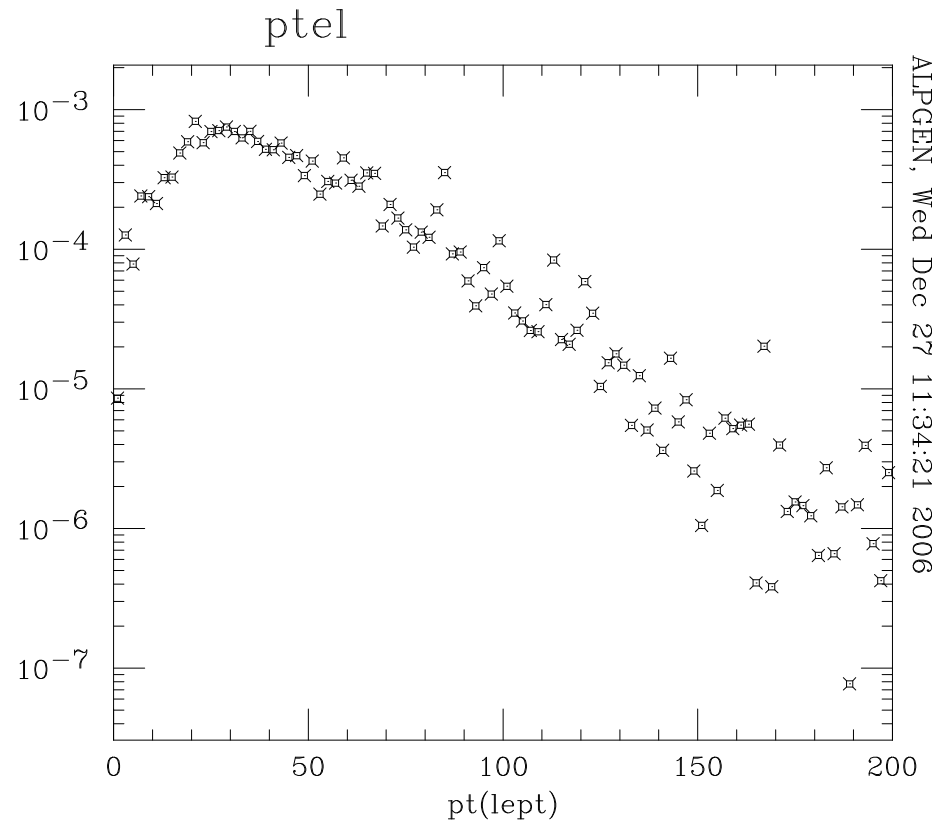
jproc= 3 total(pb): 0.00276419101+- 0.000200695731

cumulated cross-section:

avgwgt(pb)= 0.0173860873+- 0.000538143309

## wbbj.top

- The pre-defined histograms of any variable, loaded by the user in `wqqusr.f` before the run, are stored here, in the Topdrawer format. For example



## wbbj.mon

- Stores intermediate information. It is rewritten each  $10^5$  points:

```
processed=      100000.0 events
```

```
average ph-space eff=  0.409294254
```

```
avgwgt(pb)=  0.017476239+-  0.000574471343 maxwgt=  26.1532933
```

```
unwgt eff=  0.000668223264
```

```
sub-processes:
```

```
jproc= 1 total:  0.0114901936+-  0.000490463401
```

```
jproc= 2 total:  0.00322185443+-  0.000225250821
```

```
jproc= 3 total:  0.00276419101+-  0.000200695731
```



- The files `wbbj.par`, `wbbj.grid1` and `wbbj.grid2` are for internal use

*What shown up to now is enough to produce results at the parton level. If, instead, one needs to produce actual events (e.g. to perform detector simulations of more realistic studies), the following is also needed*

- The file `wbbj.wgt` contains the weighted events generated during the run. To generate unweighted events do the following 4 steps

1) Run again the code:

```
> ./wqqgen
```

Output:

Input RUN generation mode:

0: generate weighted events, no evt dumps to file

1: generate wgted events, write to file for later unweighting

2: read events from file for unweighting or processing

or documentation modes:

3: print parameter options and defaults, then stop

4: write to `par.list` parameter options and defaults, then stop

5: write to `prc.list` complete list of processes, parameter options and defaults, scale choices, PDF, etc., then stop

## 2) Input:

> 2

### Output:

Input string labeling output and input files  
(e.g. w2j to output files w2j.stat, etc.)

## 3) Input:

> wbbj

### Output:

Parameters and results written to wbbj\_unw.par  
Topdrawer plots (if any) written to wbbj\_unw.top  
Wgted events are read from wbbj.wgt  
Unwgted events are written to wbbj.unw

read in generation parameters:

ihvy = 5.

njets = 1.

ickkw = 0.

```
ptjmin = 20.  
ptbmin = 20.  
etajmax = 1.  
etabmax = 1.  
drjmin = 0.7  
drbmin = 0.7
```

Input decay params to replace defaults: type and value  
(input 'print 1' to display the list of parameter types and their current values)  
(input 'ctrl-D' to terminate the input sequence)

#### 4) Input:

```
> ctrl-D
```

#### Output:

```
starting scan/unweighting of 5450. events  
Crosssection(pb)= 0.0173861+- 0.000538143  
Generated 176 unweighted events, lum= 10123.0293pb-1
```

- The generated unweighted events can be found in the file `wbbj.unw` and are written in a format that can be easily translated to the standard *Les Houches Accord*

E. Boos *et al.*, hep-ph/0109068

- For example the first of the 176 generated unweighted events reads

```

1      1  7  0.100000E+01  0.118382E+03
2      3  0    118.206
-1     0  1   -147.774
5      2  0    12.285    -33.814    -16.966     4.700
-5     0  1    40.632    -48.437     14.085     4.700
21     3  2   -19.046     42.996     24.373     0.000
12     0  0     3.867     44.875     -0.971     0.000
-11    0  0   -37.738     -5.619    -50.088     0.001

```

## Showering and Hadronizing events (with Herwig)

- Compile Herwig by going to the appropriate directory (it takes a while...):

```
> cd herlib
```

```
> make hwuser
```

- Return to the working directory and run from the the Herwig executable file (`hwuser`). Input:

```
> cd ../wqqwork
```

```
> ../herlib/hwuser
```

Output:

HERWIG 6.510 31st Oct. 2005

Please reference: G. Marchesini, B.R. Webber,  
G.Abbiendi, I.G.Knowles, M.H.Seymour & L.Stanco  
Computer Physics Communications 67 (1992) 465

and

G.Corcella, I.G.Knowles, G.Marchesini, S.Moretti,  
K.Odagiri, P.Richardson, M.H.Seymour & B.R.Webber,  
JHEP 0101 (2001) 010

INPUT NAME OF FILE CONTAINING EVENTS  
(FOR "file.unw" ENTER "file")

- Input:

> wbbj

- The 176 unweihted events are now fully showered and hadronized. They can be analyzed on line, during the running of Herwig, from SUBROUTINE HWANAL in hwuser.f (see the Herwig Manual for more details).