whizard tutorial for physics at lepton colliders

Junping Tian (KEK)

generators used for ILC TDR physics study

- Whizard: for processes up to 6-fermion
  (http://whizard.hepforge.org)

- Physsim: for 8-fermion processes, e.g. e+e- —> ttH/ttg—>(bW)(bW)(bb)—>(bqq)(bqq)(bb)
  (http://www-jlc.kek.jp/subg/offl/physsim/)

with realistic beam spectrum, beam polarisation, spin correlations, hadronisation, etc.

but all in tree level; contribution by MadGraph community will be extremely welcome

WHIZARD in a Nutshell

WHIZARD is a universal event generator for elementary processes at colliders:

- e^+e^-: LEP and TESLA/NLC $\Rightarrow$ ILC, CLIC, ...
- pp: Tevatron $\Rightarrow$ LHC, ...

It contains

1. O'Mega: Automatic matrix elements for arbitrary elementary processes, supports SM and many BSM extensions
2. Phase-space parameterization module
3. VAMP: Generic adaptive integration and (unweighted) event generation
4. Intrinsic support or external interfaces for: Feynman rules, beam properties, cascade decays, shower, hadronization, analysis, event file formats, etc., etc.
5. Free-format steering language SINDARIN

W, Higgs, Z, And Respective Decays

Milestones

1.0 Project started around 1999: Studies for electroweak multi-particle processes at TESLA (W, Higgs, Z)

Event samples for LC studies at SLAC

1.9 Full SM w/ QCD, beam properties, SUSY/BSM, event formats

2.1 QCD shower+matching, FeynRules support, internal density-matrix formalism (cascade decays), language SINDARIN as user interface, OpenMP parallelization, . . .
(production version)

2.2 Major refactoring of internals (same user interface), event sample reweighting, inclusive processes and selective decay chains
(public alpha version)

Plan Improve ILC support; NLO + matching; improve user interface
$\Rightarrow$ adapt to specific needs of user groups

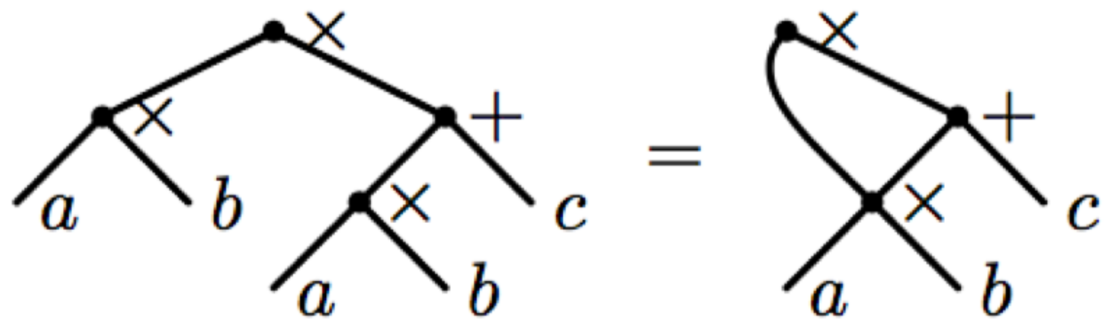
O'Mega: Optimal matrix elements

Ohl/JRR, 2001

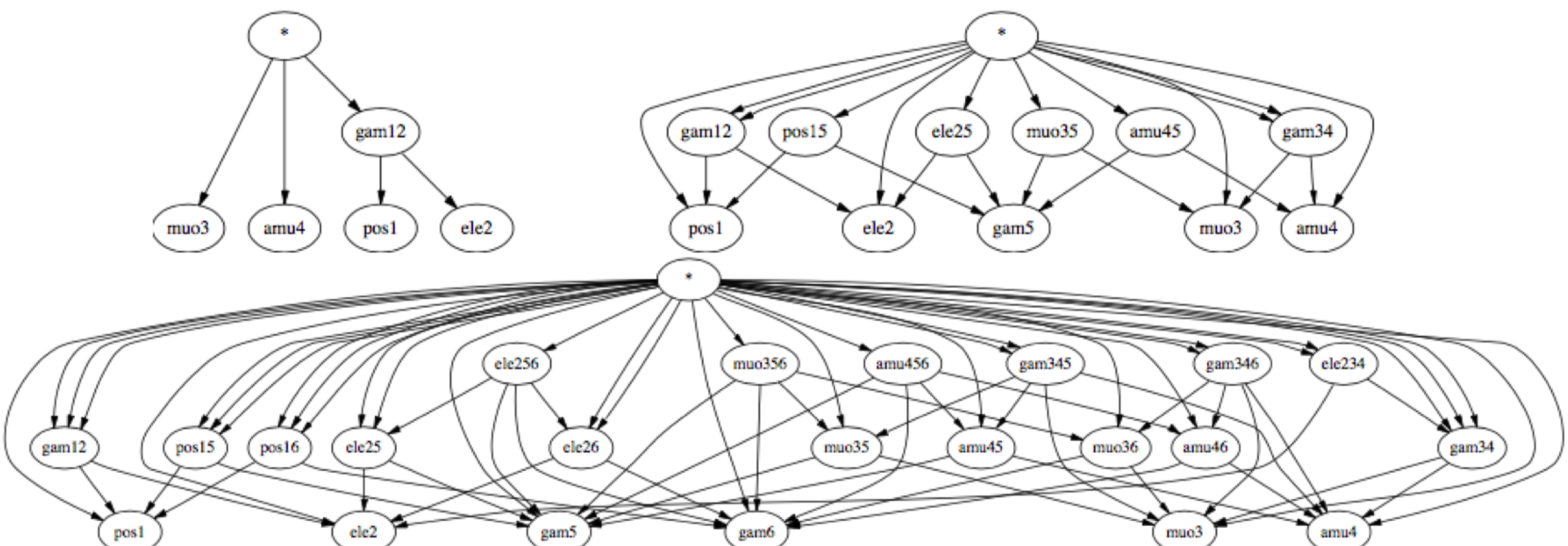
Ω

- [·.] Replace forest of tree diagrams by Directed Acyclical Graph (DAG) of the algebraic expression (including color).

$$ab(ab+c) =$$



- simplest examples: $e^+e^- \to \mu^+\mu^-$, $e^+e^- \to \mu^+\mu^-\gamma$ and $e^+e^- \to \mu^+\mu^-\gamma\gamma$



Hard matrix elements: Lorentz structures

- **Hard-coded set of Lorentz structures**
  - Purely scalar couplings: $\phi^3, \quad \phi^4$
  - Scalar couplings to vectors: $g V^\mu \phi_1 \mathrm{i}\overleftrightarrow{\partial_\mu}\phi_2, \quad \phi V^2, \quad \phi^2 V^2, \quad \frac{1}{2}\phi F_{1,\mu\nu}F_2^{\mu\nu}, \quad \frac{1}{2}\phi F_{1,\mu\nu}\tilde{F}_2^{\mu\nu}, \quad \phi(\mathrm{i}\partial_\mu V_1^\nu)(\mathrm{i}\partial_\nu V_2^\mu)$
  - Pure vector couplings: $F_{\mu\nu}F^{\mu\nu}, \quad V_1^\mu((\mathrm{i}\partial_\nu V_2^\rho)\mathrm{i}\overleftrightarrow{\partial_\mu}(\mathrm{i}\partial_\rho V_3^\nu)), \quad gF_1^{\mu\nu}F_{2,\nu\rho}F_{3,}{}^{\rho}{}_{\mu},$
    $g/2 \cdot \epsilon^{\mu\nu\lambda\tau}F_{1,\mu\nu}F_{2,\tau\rho}F_{3,}{}^{\rho}{}_{\lambda}$
  - Fermionic couplings to scalars:
    $g_S\bar{\psi}_1 S\psi_2, \quad g_P\bar{\psi}_1 P\gamma_5\psi_2, \quad \bar{\psi}_1\phi(g_S + g_P\gamma_5)\psi_2, \quad g_L\bar{\psi}_1\phi(1-\gamma_5)\psi_2, \quad g_R\bar{\psi}_1\phi(1+\gamma_5)\psi_2,$
    $g_L\bar{\psi}_1\phi(1-\gamma_5)\psi_2 + g_R\bar{\psi}_1\phi(1+\gamma_5)\psi_2$
  - Fermionic couplings to vectors:
    $g_V\bar{\psi}_1 \not{V}\psi_2, \quad g_A\bar{\psi}_1\gamma_5\not{V}\psi_2, \quad \bar{\psi}_1\not{V}(g_V - g_A\gamma_5)\psi_2, \quad g_L\bar{\psi}_1\not{V}(1-\gamma_5)\psi_2, \quad g_R\bar{\psi}_1\not{V}(1+\gamma_5)\psi_2,$
    $g_L\bar{\psi}_1\not{V}(1-\gamma_5)\psi_2 + g_R\bar{\psi}_1\not{V}(1+\gamma_5)\psi_2$
  - Fermionic couplings to tensors: $g_T T_{\mu\nu}\bar{\psi}_1[\gamma^\mu, \gamma^\nu]_-\psi_2$
  - Tensor couplings to vectors:
    $T^{\mu\nu}(V_{1,\mu}V_{2,\nu} + V_{1,\nu}V_{2,\mu}), \quad T^{\alpha\beta}(V_1^\mu \mathrm{i}\overleftrightarrow{\partial}_\alpha \mathrm{i}\overleftrightarrow{\partial}_\beta V_{2,\mu},$
    $T^{\alpha\beta}(V_1^\mu \mathrm{i}\overleftrightarrow{\partial}_\beta(\mathrm{i}\partial_\mu V_{2,\alpha}) + V_1^\mu \mathrm{i}\overleftrightarrow{\partial}_\alpha(\mathrm{i}\partial_\mu V_{2,\beta})), \quad T^{\alpha\beta}((\mathrm{i}\partial^\mu V_1^\nu)\mathrm{i}\overleftrightarrow{\partial}_\alpha \mathrm{i}\overleftrightarrow{\partial}_\beta(\mathrm{i}\partial_\nu V_{2,\mu}))$
  - Gravitino couplings: $\bar{\psi}\gamma^\mu S\psi_\mu, \quad \bar{\psi}\gamma^\mu \not{k}_S S\psi_\mu, \quad \bar{\psi}\gamma^\mu\gamma^5 P\not{k}_P\psi_\mu, \quad \bar{\psi}\gamma^5\gamma^\mu[\not{k}_V, \not{V}]\psi_\mu$ etc.
- Completely general Lorentz structures:
  work in progress, to appear in version 2.2

J. Reuter @ KEK, 2013

WHIZARD – Overview over BSM Models

MODEL TYPE	with CKM matrix	trivial CKM
QED with e, μ, τ, γ	–	QED
QCD with d, u, s, c, b, t, g	–	QCD
Standard Model	**SM_CKM**	**SM**
SM with anomalous gauge coupl.	**SM_ac_CKM**	**SM_ac**
SM with anomalous top coupl.	**SMtop_CKM**	**SMtop**
SM with K matrix	—	SM_KM
2HDM	2HDM_CKM	2HDM
MSSM	MSSM_CKM	MSSM
MSSM with gravitinos	—	MSSM_Grav
NMSSM	NMSSM_CKM	NMSSM
extended SUSY models	—	PS/E/SSM
Littlest Higgs	—	Littlest
Littlest Higgs with ungauged $U(1)$	—	Littlest_Eta
Littlest Higgs with T parity	—	Littlest_Tpar
Simplest Little Higgs (anomaly-free)	—	Simplest
Simplest Little Higgs (universal)	—	Simplest_univ
3-site model	—	Threeshl
UED	—	UED
SM with Z'	—	Zprime
SM with gravitino and photino	—	GravTest
Augmentable SM template	—	Template

new models easily: FeynRules interface Christensen/Duhr/Fuks/JRR/Speckner, 1010.3251

Interface to SARAH in the SUSY Toolbox Staub, 0909.2863; Ohl/Porod/Speckner/Staub, 1109.5147

comment on Whizard-1.95

- Whizard 2 is being developed actively (latest 2.2.8, released in this week), and provides many new features and improvements, e.g. new steering language, 8f doable, speeding by parallel, etc.; but some interfaces to ILC simulation are not ready; will be ready in short term

- Whizard 1.95 is still the standard one as having been widely used by ILC physics community for many years, by this tutorial as well…

tutorial set up

- simplest way: get a copy of the disk image of virtual machine (~6 GB); get it installed in your virtualbox; then you will already have all the needed libraries and also examples for this tutorial.

#if get that virtual machine started successfully, first thing you may would like to do is **change the keyboard layout:** System —>Preference—>Keyboard—>layout, set default your favourite (currently Japanese layout)

tutorial set up

- alternatively: make available following libs

```
######### prerequisites to install whizard######
## following libraries are necessary
# 1. GNU Make
# 2. PERL 5
# 3. Fortran 95
# 4. Fortran 77
# 5. C compiler
# 6. Objective Caml (e.g. ocaml-3.11.2)

## following libraries are only needed for more detailed analysis
# 1. Pythia    (for fragmentation)
# 2. STDHEP    (for output events)
# 3. Tauola    (for proper decay of tau regarding polarisations)
# 4. Latex     (for plotting Feynman Diagrams)
# 5. MetaPost  (for plotting histograms)
# 6. PDFLIB    (for parton distribution functions, from CERNLIB)
```

http://www-jlc.kek.jp/~tianjp/whizard_tutorial_MGS2015.tar.gz

get tutorial started

- cd $HOME/whizard_tutorial_MGS2015 ; ls ;

```
README          example02/      packages/
example00/      example03/      processes/
example01/      example04/      setup_for_whizard.sh
```

- emacs README ;

- source setup_for_whizard.sh ;

example01

$$e^+e^- \to \mu^+\mu^- H$$

- tar zxvf packages/whizard-1.95.tgz -C example01 ;
- cd example01 ; ./configure ;
- cp ../processes/whizard.example01.prc conf/whizard.prc ;
- make prg install ;

example01

$$e^+e^- \to \mu^+\mu^- H$$

- cd results ;
- cp ../../processes/whizard.example01.in whizard.in ;
- cp ../../processes/whizard.example01.cut5 whizard.cut5
- ./whizard ;

#find cross sections and summary in “whizard.out”

#10K events should have been saved to “whizard.evt”

#visualize the Feynman diagrams in this process

- make channels ; evince “whizard-channels.ps" ;

#visualize some kinematic distributions

- make plots ; evince “whizard-plots.ps" ;

example01

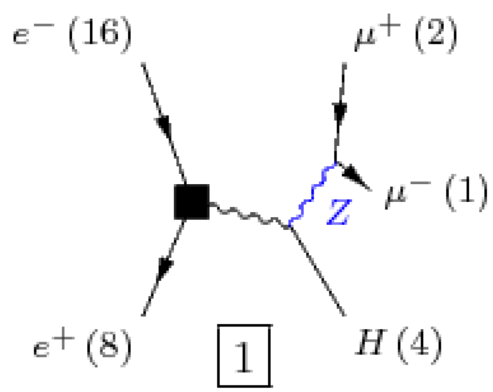
$$e^+e^- \to \mu^+\mu^- H$$

WHIZARD phase space channels

Process: e2e2h_o ($e^-e^+ \to \mu^-\mu^+ H$)
Color code: resonance, t-channel, radiation, infrared, collinear, external/off-shell

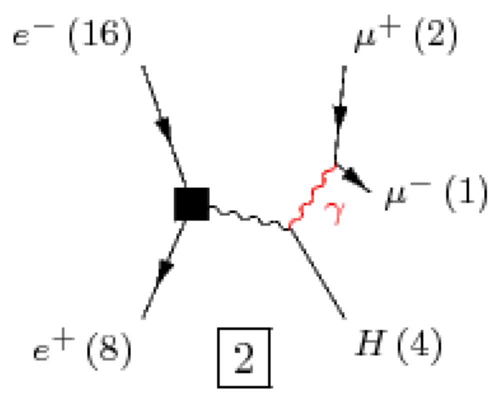
Grove 1

Multiplicity: 2
Resonances: 1
Log-enhanced: 0
t-channel: 0



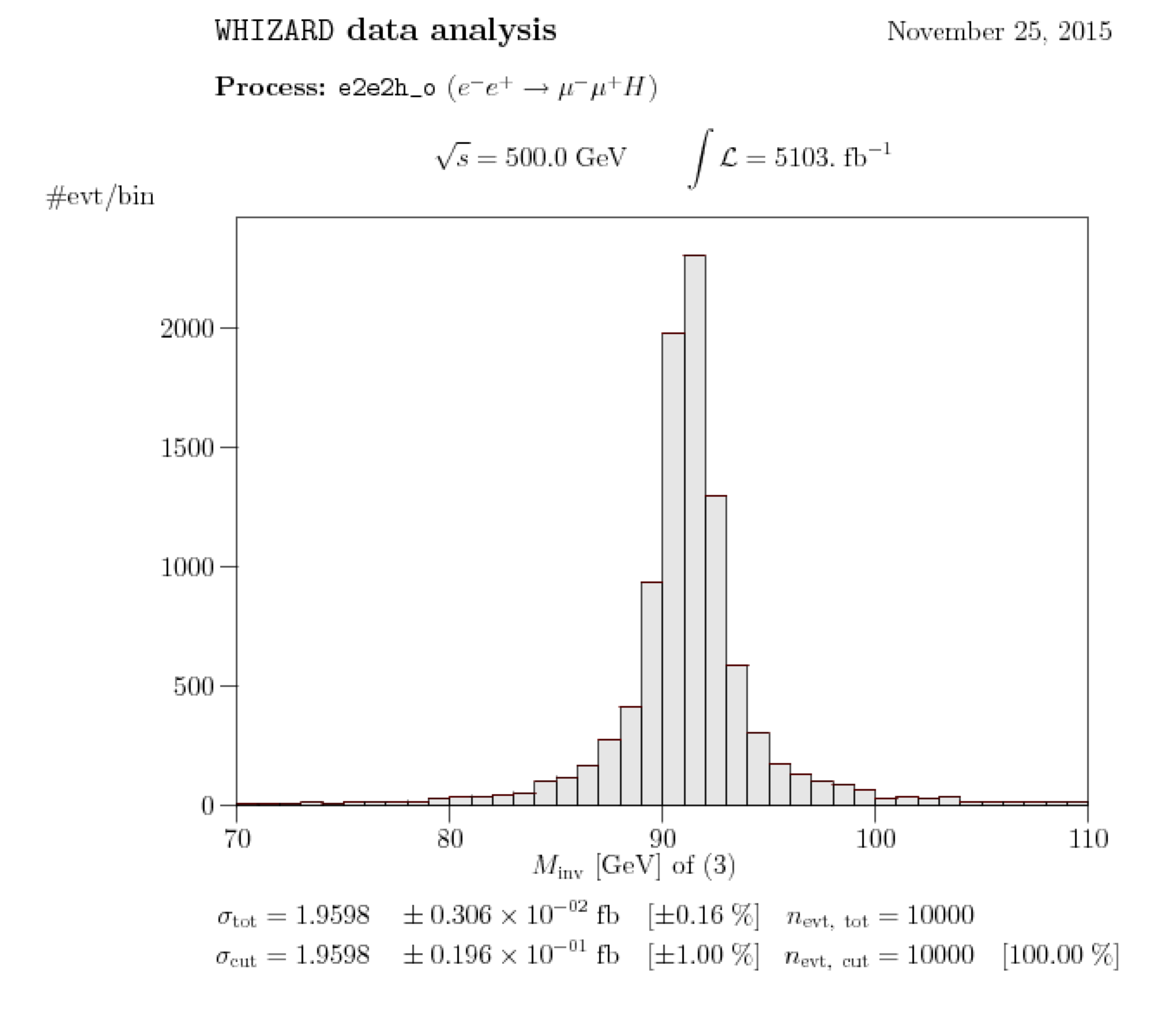
Grove 2

Multiplicity: 3
Resonances: 0
Log-enhanced: 1
t-channel: 0



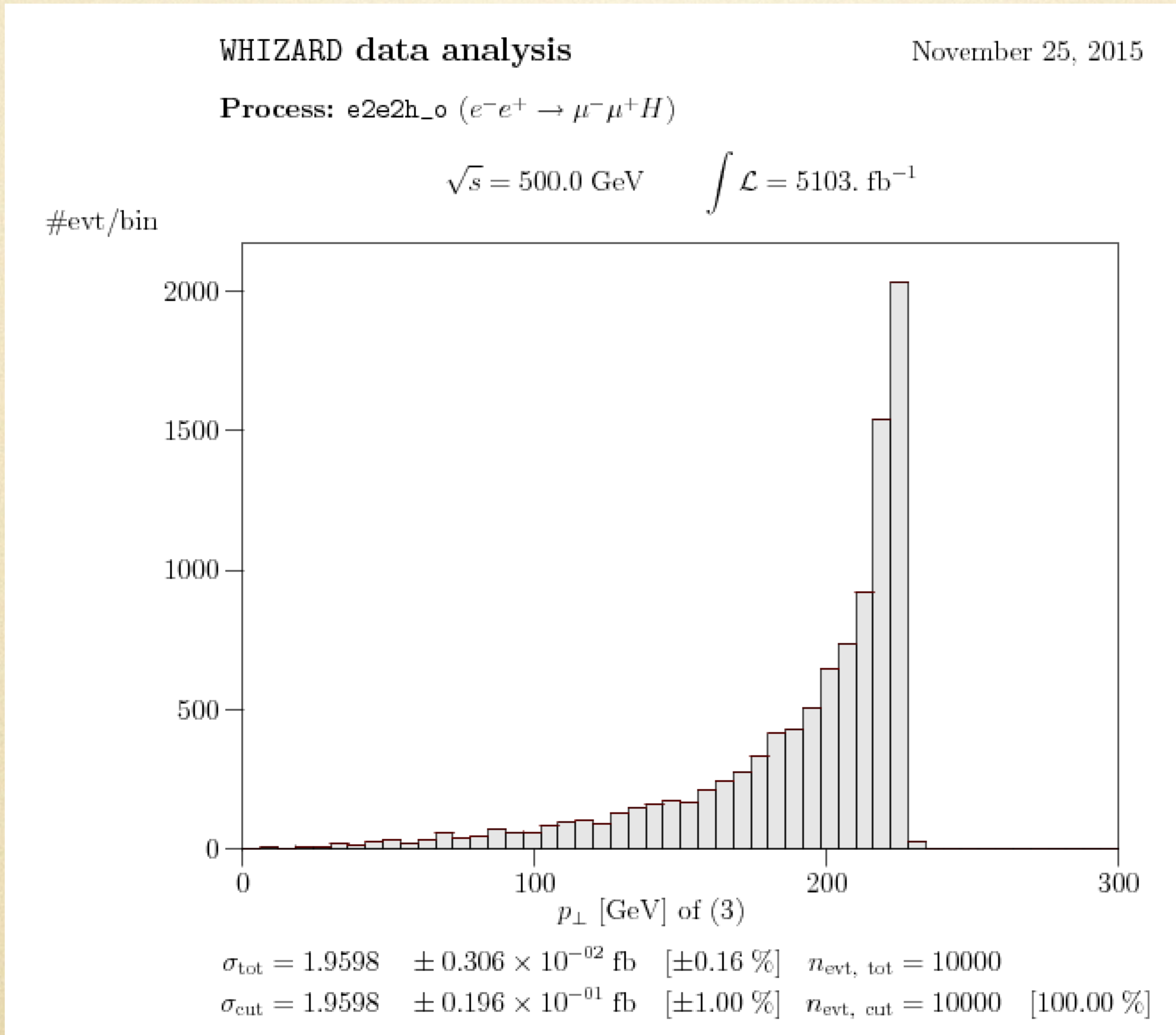
example01

$$e^+e^- \to \mu^+\mu^- H$$



example01

$$e^+e^- \to \mu^+\mu^- H$$



example01: ingredients to run a process by whizard

- define a process in “whizard.prc”
- configure model parameters in “whizard.in”
- histogram and cut in “whizard.cut5”

#all located in the “example01/results/”

define a process in “whizard.prc”

```
# The selected model (O'Mega)
model    SM_CKM

# Processes for example01
# (Methods: chep=CompHEP, mad=MadGraph, omega=O'Mega)
# (Options: s=selected diagrams, number=QCD order [Madgraph])
#          f=fudged width [O'Mega]
#
# Tag            In      Out              Method  Option
#===================================================
##############################################################
###########
#
e2e2h_o     e1,E1  e2,E2,h                omega   w:c,c
```

#all the particle names are defined in “${Model}.mdl”, e.g. “example01/conf/models/SM_CKM.mdl”;

particle: e1,e2,e3,n1,n2,n3,u,d,c,s,t,b

anti-particle: E1,E2,E3,N1,N2,N3,U,D,C,S,T,B

W+,W-,Z,h,a,g

```
# WHIZARD configuration file

# The selected model
model   SM

# Processes
#  Methods: chep=CompHEP, mad=MadGraph, omega=O'Mega, test=trivial)
#  Options: s      selected diagrams (CompHEP/MadGraph)
#           r      restricted intermediate state (O'Mega)
#           c      apply exact color algebra (O'Mega)
#           n:XXX  coupling order (MadGraph)
#           w:XXX  width scheme (O'Mega)
#           p      transfer polarization (test)
#           u      unit matrix element (test)
#
# Tag             In      Out              Method  Option
#=====================================================
ee_c            e1,E1   e1,E1            chep
ee_m            e1,E1   e1,E1            mad
ee_o            e1,E1   e1,E1            omega
ww_c            e1,E1   W+,W-            chep
ww_m            e1,E1   W+,W-            mad
ww_o            e1,E1   W+,W-            omega
zh_c            e1,E1   Z,H              chep
zh_m            e1,E1   Z,H              mad
zh_o            e1,E1   Z,H              omega
nnh_c           e1,E1   n1,N1,H          chep
nnh_m           e1,E1   n1,N1,H          mad
nnh_o           e1,E1   n1,N1,H          omega
nnbb_m          e1,E1   n1,N1,b,B        mad
nnbb_o          e1,E1   n1,N1,b,B        omega
```

configure process parameters in “whizard.in”

```
&process_input
process_id = "e2e2h_o"
sqrts = 500
luminosity = 10
/
```

#specify process, ecm, lumi, etc.

```
&integration_input
/
```

#control integration accuracy, seed, etc.

```
&simulation_input
n_events = 10000
/
```

#set no. evts, evt. format, fragmentation, etc.

```
&diagnostics_input
/
```

#set time limit, etc.

```
&parameter_input
Mmu = 0
MH = 125
wH = 0.0043
/
```

#set particle mass, coupling constant, etc.

```
&beam_input
/
```

#set beam spectrum, polarisation, ISR, etc.

```
&beam_input
/
```

##see complete config in example04

histogram and cut in “whizard.cut5”

```
e2e2h_o     e1,E1  e2,E2,h                 omega
```

binary code 8 16 1 2 4

```
# Analysis configuration file for the process ee -> mu mu h
process e2e2h_o
histogram M of 3 within 70 110 nbin 40
and
histogram E of 4 within 100 300 nbin 50
and
histogram PT of 3 within 0 300 nbin 50
and
histogram CTA of 3 within -1 1 nbin 50
and
histogram CTA of 1 within -1 1 nbin 50
and
histogram CTA of 2 within -1 1 nbin 50
```

cut PT of 4 within 0 150

cut M of 3 within 80 100 or 180 200 or 500 99999

example00

#compare results using matrix elements
by **Omega, ComHEP, MadGraph**

- tar zxvf packages/whizard-1.95.tgz -C example00 ;
- cd example00 ;
- cp ../processes/config.example00.site config.site ;
- ./configure ;
- make test ; make test-QED ; make test-QCD

#there’s a typo in README; order 1.3 and 1.4 should be swapped

example00

#compare results using matrix elements
by **Omega, ComHEP, MadGraph**

$e^+e^- \to \nu\bar{\nu}H$

```
!=============================================================================
! Summary (all processes):
!-----------------------------------------------------------------------------
! Process ID      Integral[fb]  Error[fb]   Err[%]        Frac[%]
!-----------------------------------------------------------------------------
  nnh_c           8.4711467E+01  3.52E-01    0.42          33.35
  nnh_m           8.4435645E+01  3.40E-01    0.40          33.24
  nnh_o           8.4890431E+01  3.53E-01    0.42          33.42
!-----------------------------------------------------------------------------
  sum             2.5403754E+02  6.04E-01    0.24         100.00
!=============================================================================
```

$e^+e^- \to \mu^+\mu^-$

```
Running WHIZARD for process ee_e2: e1,E1 -> e2,E2
   ee_e2_o         3.4821868E+02  3.20E+00    0.92          33.33
   ee_e2_m         3.4821868E+02  3.20E+00    0.92          33.33
   ee_e2_c         3.4821868E+02  3.20E+00    0.92          33.33
```

$u\bar{u} \to t\bar{t}$

```
Running WHIZARD for process uu_u3: u,U -> t,T
 **uu_u3_o         1.7958449E+04  6.86E+01    0.38          33.33
   uu_u3_m         1.7958450E+04  6.86E+01    0.38          33.33
   uu_u3_c         1.7958450E+04  6.86E+01    0.38          33.33
```

example02

$$e^+e^- \to \nu_e\bar{\nu}_e H \qquad\qquad e^+e^- \to e^+e^- H$$

- tar zxvf packages/whizard-1.95.tgz -C example02 ;
- cd example02 ; ./configure ;
- cp ../processes/whizard.example02.prc conf/whizard.prc ;
- make prg install ;
- cd results ;
- cp ../../processes/whizard.example02.in whizard.in ;
- cp ../../processes/whizard.example02.cut5 whizard.cut5
- ./whizard ;
- make channels ; evince “whizard-channels.ps" ;
- make plots ; evince “whizard-plots.ps" ;

example02: select Feynman diagrams in “whizard.prc”

$$e^+e^- \to \nu_e \bar{\nu}_e H \qquad e^+e^- \to e^+e^- H$$

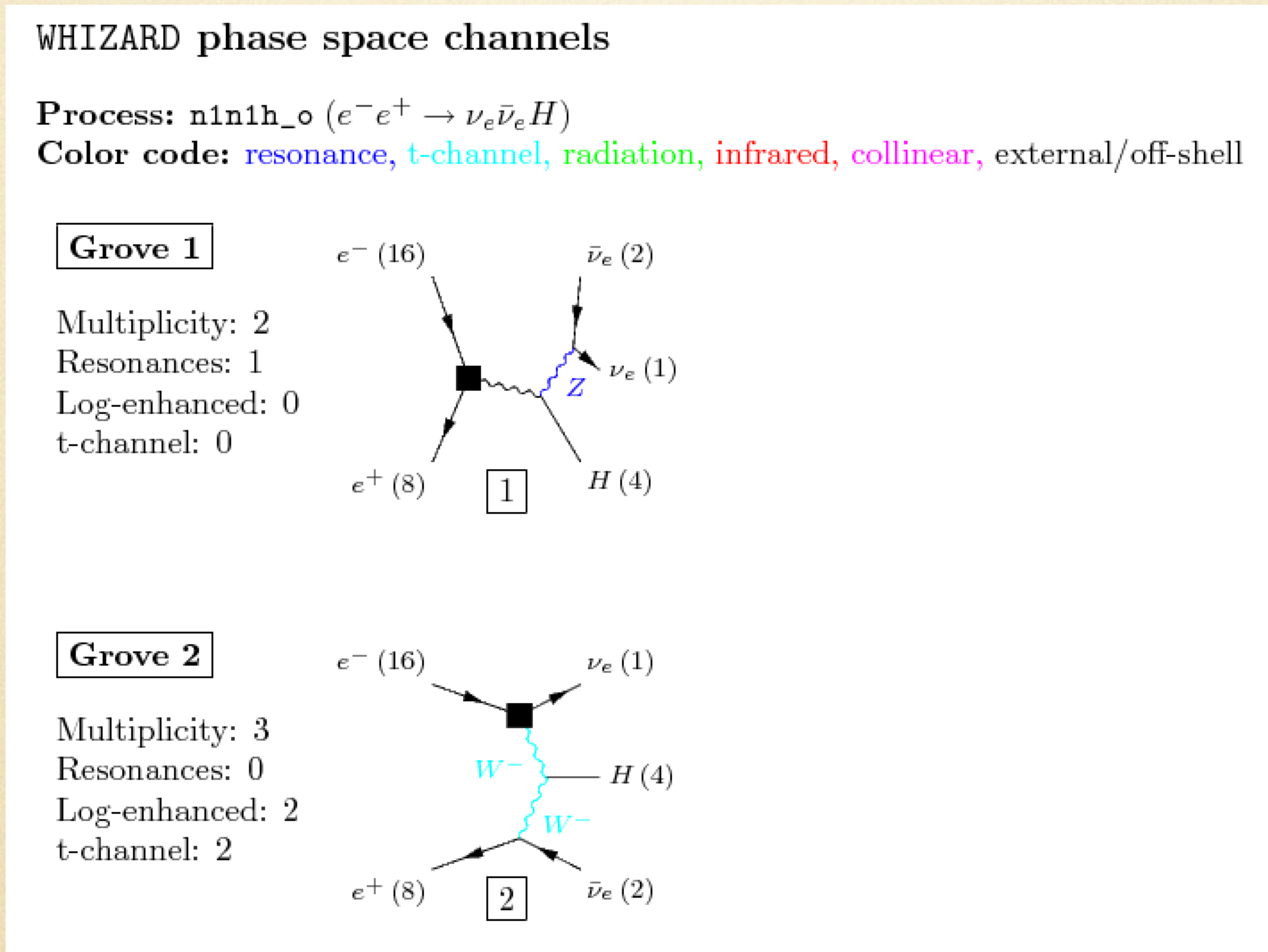
```
# Tag              In      Out                    Method   Option
#=====================================================
##################################################################
#
n1n1h_o         e1,E1   n1,N1,h           omega   w:c,c
n1n1h_s_o       e1,E1   n1,N1,h           omega   w:c,c,r:3+4~Z
n1n1h_t_o       e1,E1   n1,N1,h           omega   w:c,c,r:1+3~W- && 2+4~W+

e1e1h_o         e1,E1   e1,E1,h           omega   w:c,c
e1e1h_s_o       e1,E1   e1,E1,h           omega   w:c,c,r:3+4~Z
e1e1h_t_o       e1,E1   e1,E1,h           omega   w:c,c,r:1+3~Z && 2+4~Z
```

#use option “r” to restrict the intermedia state

#note the particle number in “r” is different with binary code

example02: select Feynman diagrams in “whizard.prc”

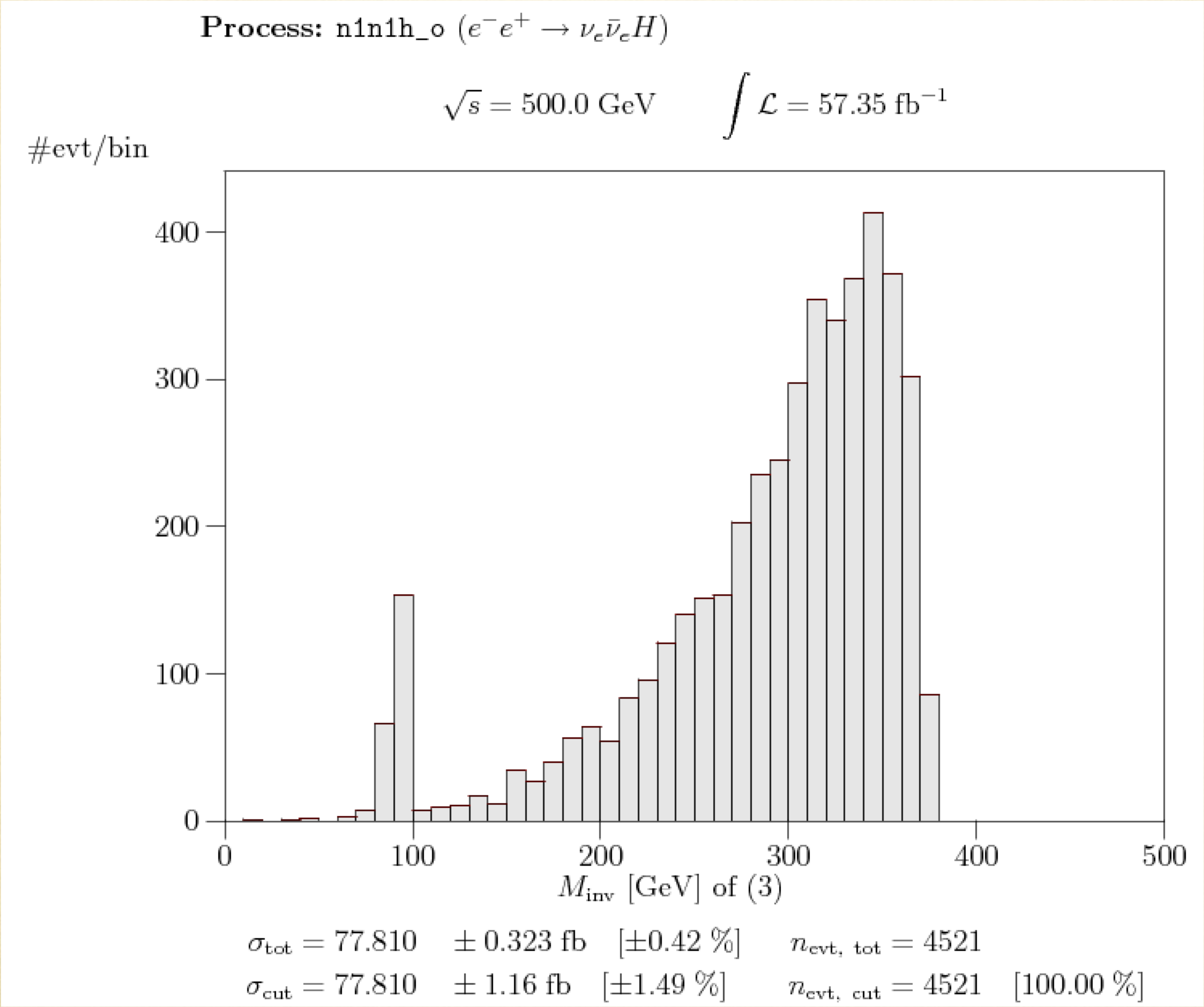
$$e^+e^- \rightarrow \nu_e\bar{\nu}_e H$$



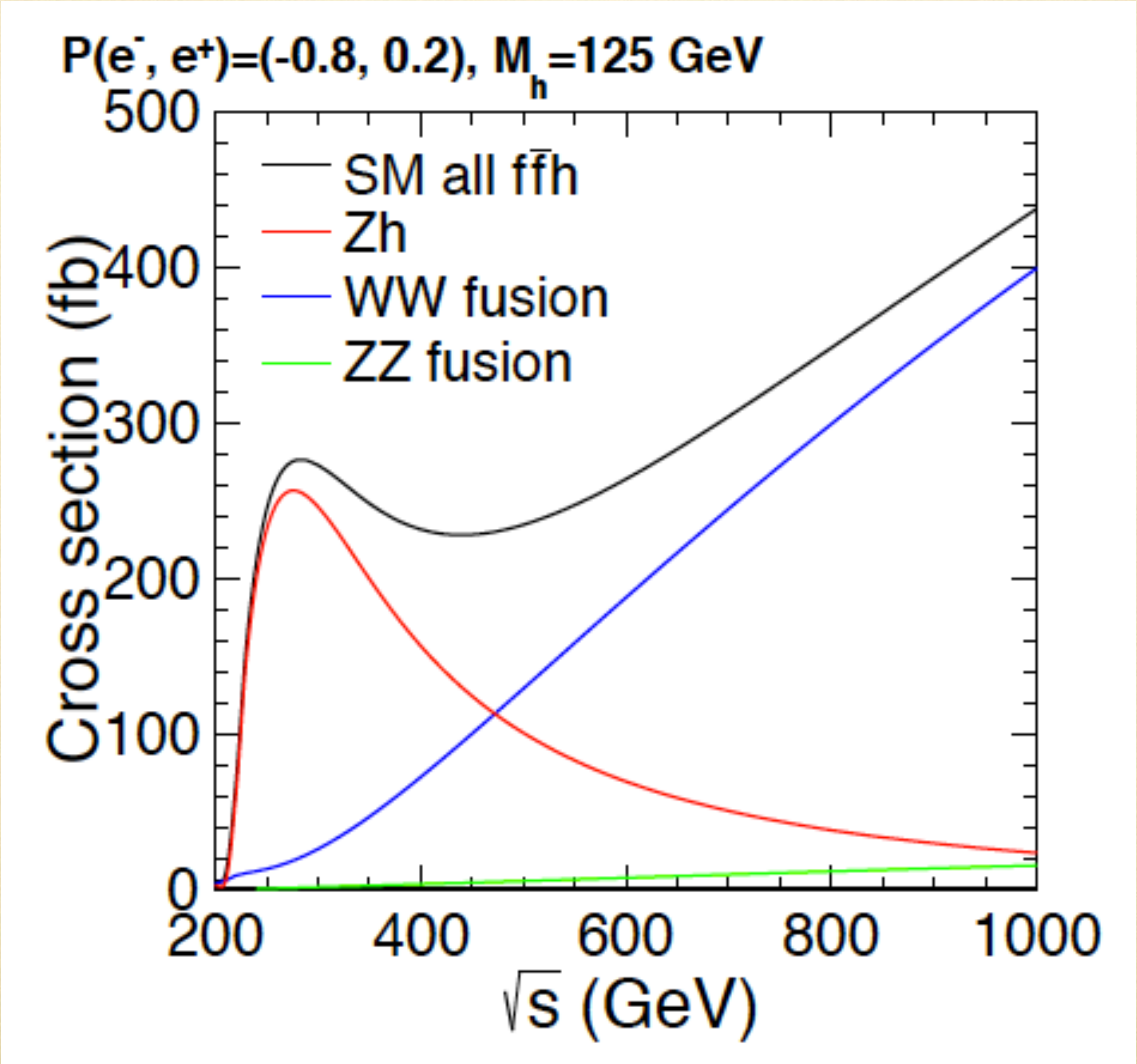
#be careful when use restriction; do not violate gauge invariance, neither ignore interference

example02

$$e^+e^- \to \nu_e \bar{\nu}_e H$$



homework (A): use whizard to reproduce following plot



example03: on-shell and off-shell Higgs decay

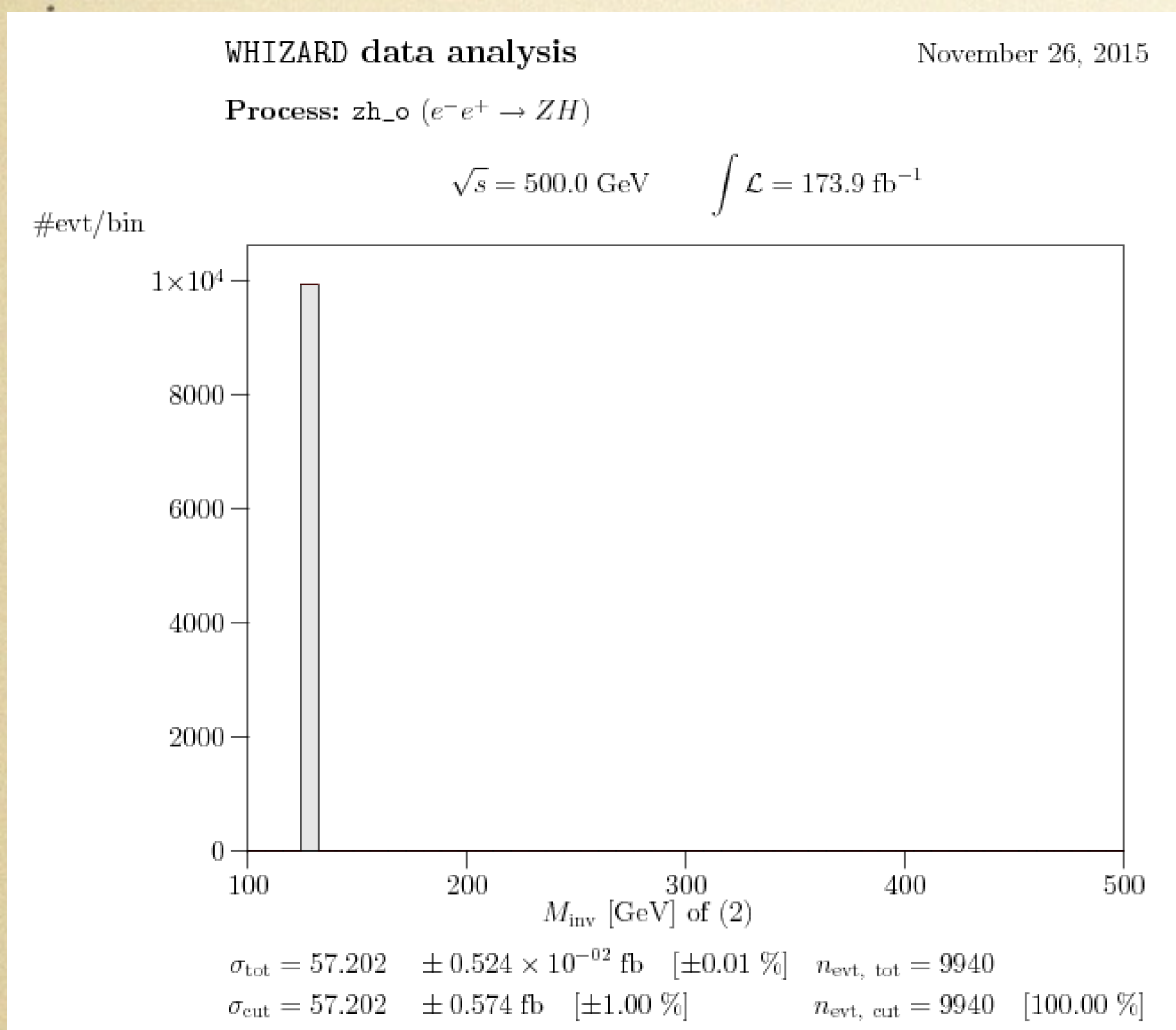
$$e^+e^- \to ZH \qquad\qquad e^+e^- \to ZH \to Z(W^+W^-)$$

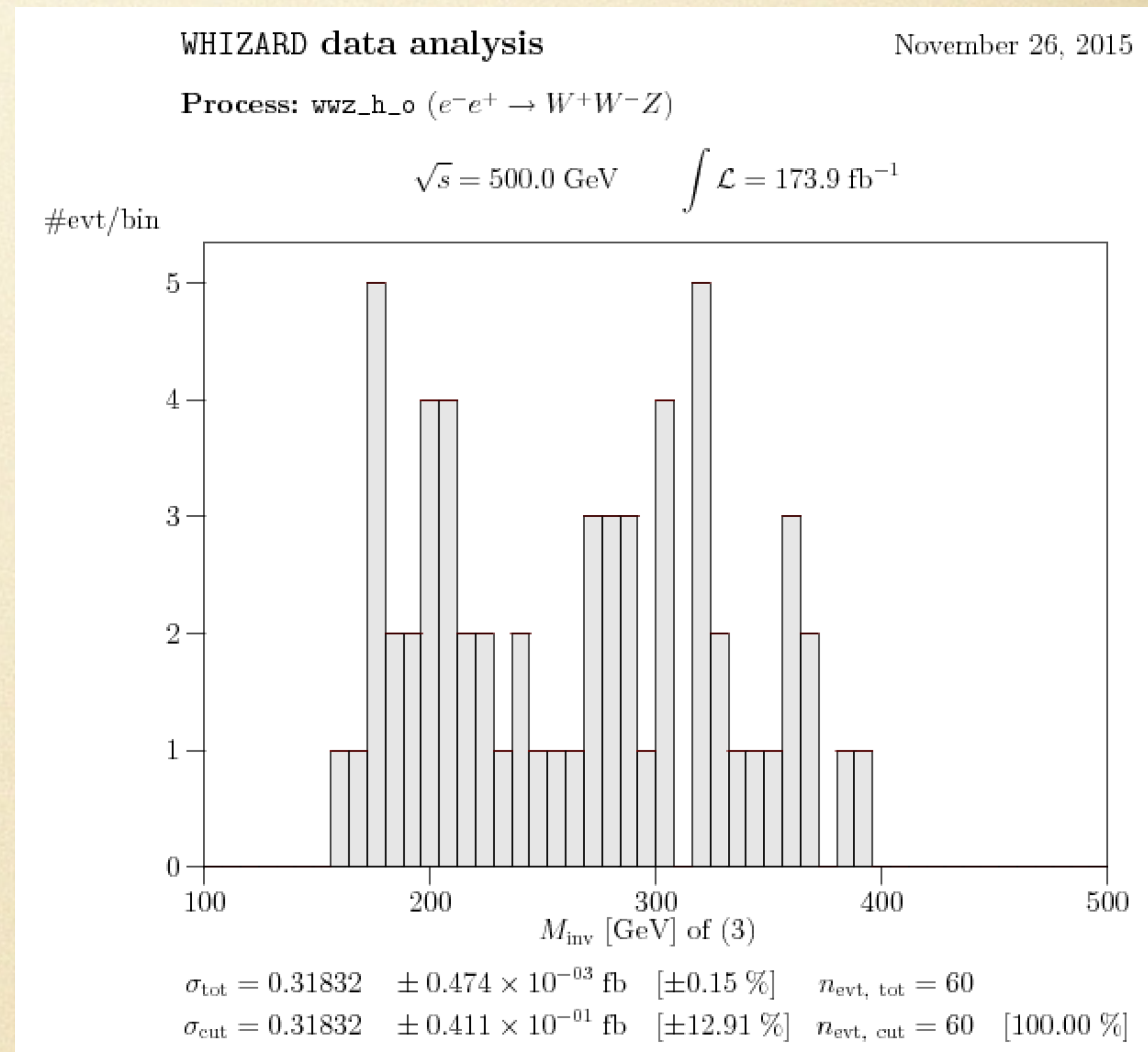
```
# Tag                 In       Out                    Method   Option
#=======================================================
##################################################################
#
zh_o                  e1,E1   Z,h                     omega   w:c,c
wwz_h_o               e1,E1   W+,W-,Z                 omega   w:c,c,r:3+4~h
```

#same procedure to run as example01/02; see README

example03: on-shell and off-shell Higgs decay

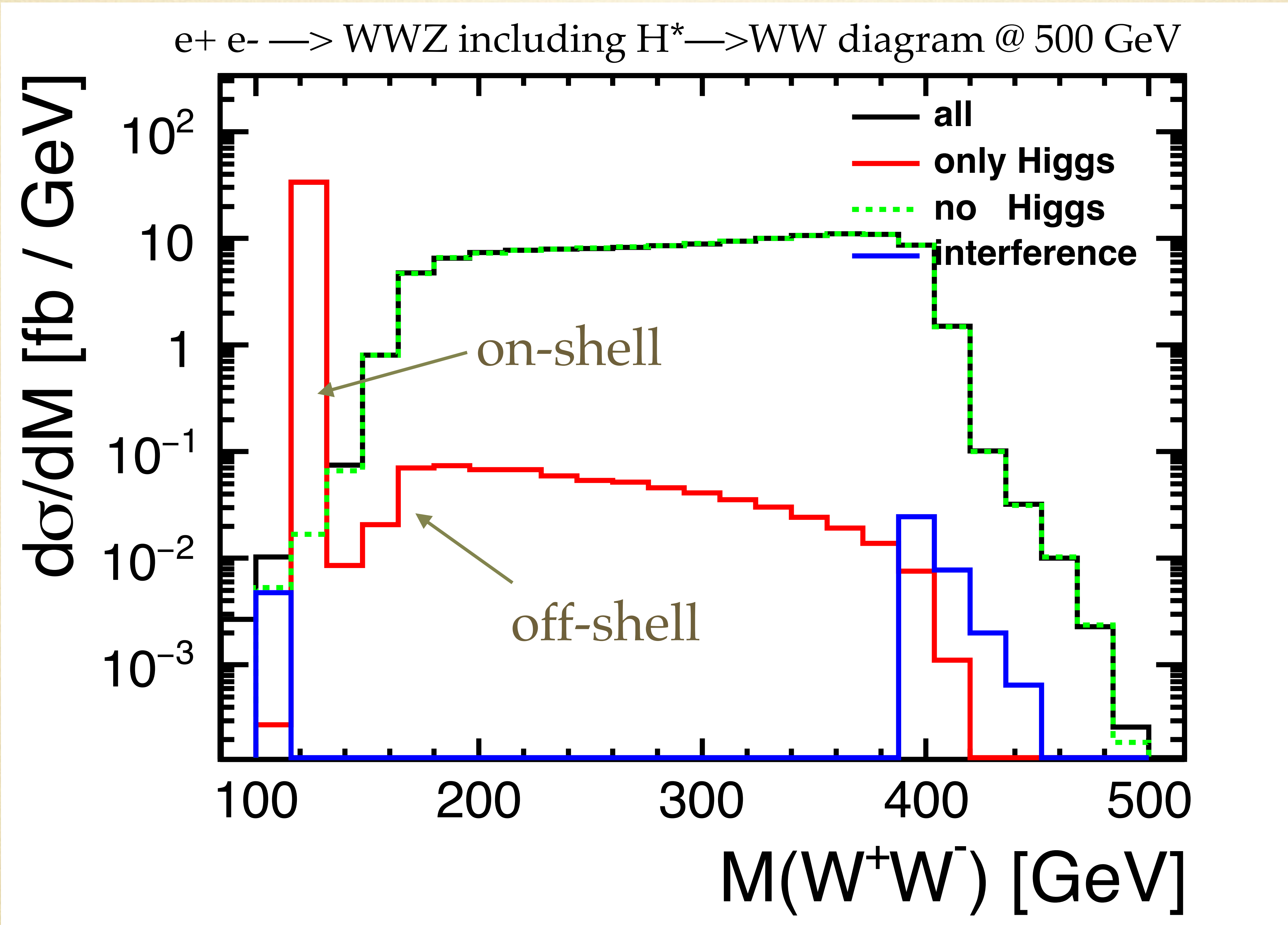
$$e^+e^- \to ZH \qquad\qquad e^+e^- \to ZH \to Z(W^+W^-)$$





#off-shell contribution / on-shell ~ 3%

homework (B): e+e- —>WWZ including both on-shell and off-shell Higgs contribution



example04: full ILC setup

#more automatic, detailed instruction see example04/README

```
# The selected model (O'Mega)
model    SM_CKM

# Tag             In      Out                 Method  Option
#=====================================================
#
alias f u:d:s:c:b:e1:e2:e3:n1:n2:n3
alias F U:D:S:C:B:E1:E2:E3:N1:N2:N3

# Tree ffh process
ffh_o       e1,E1  f,F,h                      omega  w:c,c
ffhh_o      e1,E1  f,F,h,h                    omega  w:c,c
```

#note features of using alias to include all possible processes

example04: full ILC setup

#e.g. enable beam spectrum

```
&process_input
process_id = ""
sqrts = 500
luminosity = 1000
polarized_beams = T
structured_beams = T
beam_recoil = T
/
```

```
&beam_input
particle_name = 'e1'
polarization = 1.0 0.0
USER_spectrum_on = T
USER_spectrum_mode = 21
ISR_on = T
ISR_alpha = 0.0072993
ISR_m_in = 0.000511
EPA_on = F
EPA_alpha = 0.0072993
EPA_m_in = 0.000511
EPA_mX = 4.
EPA_Q_max = 4.
/
```

example04: full ILC setup

#e.g. tune fragmentation parameters, specify Higgs decay branching ratios, set output event format (e.g. STDHEP)

```
 &simulation_input
 n_events = 10000
 write_events_raw = F
 keep_beam_remnants = T
 fragment = T
 fragmentation_method = 3
! OPAL tune
  pythia_parameters = "PMAS(25,1)=125.0; PMAS(25,2)=0.0043;
  MSTJ(41)=2; MSTU(22)=20; MSTJ(28)=2;
  PARJ(21)=0.40000; PARJ(41)=0.11000; PARJ(42)=0.52000; PARJ(81)=0.25000;
  PARJ(82)=1.90000; MSTJ(11)=3; PARJ(54)=-0.03100; PARJ(55)=-0.00200;
  PARJ(1)=0.08500; PARJ(3)=0.45000; PARJ(4)=0.02500; PARJ(2)=0.31000;
  PARJ(11)=0.60000; PARJ(12)=0.40000; PARJ(13)=0.72000; PARJ(14)=0.43000;
  PARJ(15)=0.08000; PARJ(16)=0.08000; PARJ(17)=0.17000; MSTP(3)=1;
  MWID(25)=2;
  BRAT(212)=0.00044;BRAT(213)=0.0268;BRAT(214)=0.578;BRAT(219)=0.000221;
  BRAT(220)=0.0637;BRAT(222)=0.0856;BRAT(223)=0.0023;BRAT(224)=0.00155;
  BRAT(225)=0.0267;BRAT(226)=0.216"
! MDME(219,1)=0;  To suppress h-> mu mu decay
  write_events = T
  write_events_format = 20
  bytes_per_file = 500000000
 /
```

#based on STDHEP events, use similar analysis method as what used for MadGraph based analysis, or SGV/ilcsoft which are commonly used by ILC community

homework (C): plot recoil mass using Z—>μμ by taking into account realistic beam spectrum