The Event Generator WHIZARD

Jürgen R. Reuter, DESY

WHIZARD: Some (technical) facts

WHIZARD v2.2.6 (02.05.2015) http://whizard.hepforge.org

<whizard@desy.de>

WHIZARD Team: *Wolfgang Kilian, Thorsten Ohl, JRR*

Bijan Chokoufé/Marco Sekulla/Christian Weiss/Florian Staub + *2 Master + 2 PhD (soon)*

(some losses: C. Speckner [software engineering], F. Bach [ESA Space Defense], S. Schmidt [Philosophy])

Publication: EPJ C71 (2011) 1742 (and others for O'Mega, Interfaces, color flow formalism)

WHIZARD: Some (technical) facts

WHIZARD v2.2.6 (02.05.2015)   http://whizard.hepforge.org

<whizard@desy.de>

WHIZARD Team: *Wolfgang Kilian, Thorsten Ohl, JRR*

Bijan Chokoufé/Marco Sekulla/Christian Weiss/Florian Staub + *2 Master + 2 PhD (soon)*

(some losses: C. Speckner [software engineering], F. Bach [ESA Space Defense], S. Schmidt [Philosophy])

Publication: EPJ C71 (2011) 1742 (and others for O'Mega, Interfaces, color flow formalism)

2nd WHIZARD Workshop Würzburg, 03/2015

WHIZARD: Some (technical) facts

WHIZARD v2.2.6 (02.05.2015) http://whizard.hepforge.org

<whizard@desy.de>

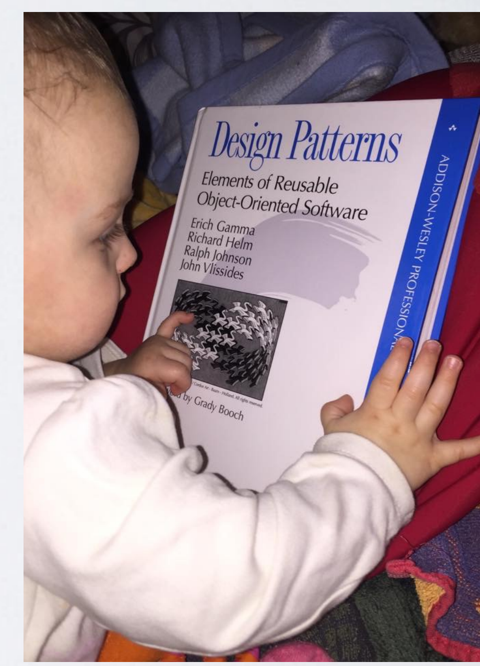
WHIZARD Team: *Wolfgang Kilian, Thorsten Ohl, JRR*

Bijan Chokoufé/Marco Sekulla/Christian Weiss/Florian Staub + *2 Master + 2 PhD (soon)*

(some losses: C. Speckner [software engineering], F. Bach [ESA Space Defense], S. Schmidt [Philosophy])

Publication: EPJ C71 (2011) 1742 (and others for O'Mega, Interfaces, color flow formalism)

2nd WHIZARD Workshop Würzburg, 03/2015



support junior developers

The WHIZARD Event Generator

- Universal event generator for lepton and hadron colliders
- Modular package:
  - Phase space parameterization (resonances, collinear emission, Coulomb etc.)
  - O'Mega optimized matrix element generator (tree level, NLO external)
  - VAMP: adaptive multi-channel Monte Carlo integrator
  - CIRCE1/2: generator/simulation tool for lepton collider beam spectra
  - Modules for beam structure, parton shower, matching/merging, event formats, analysis, cascade decays, polarized initial/final states, [NLO subtractions] etc.
  - Interfaces to external packages for Feynman rules, hadronization, tau decays, event formats, analysis, jet clustering etc.
  - SINDARIN: free-format steering language for all inputs (!)

Talk concentrates
on NEW features
and
current developments/
(near) future plans

WHIZARD: Installation and Run

- Download: http://www.hepforge.org/archive/whizard/whizard-2.2.6.tar.gz
- Unpack it, intended to be installed in `/usr/local` (or locally)
- Create `build` directory and do `./configure`
- `make`, `[ make check ]`, `make install`
- Working directory: create `SINDARIN` steering file `<input>.sin`
- Working directory: run `whizard <input>.sin`
- Supported event formats: LHA, StdHep, LHEF (i-iii), HepMC, LCIO, div. ASCII
- Interfaces to external packages: `FastJet, GoSam, GuineaPig(++), HepMC, HOPPET, LCIO, LHAPDF(4/5/6), LoopTools, OpenLoops, PYTHIA6, [PYTHIA8], StdHep`

```
PASS: circe2_2.run
PASS: ewa_1.run
PASS: ewa_2.run
PASS: ewa_3.run
PASS: ewa_4.run
PASS: ilc.run
PASS: gaussian_1.run
PASS: gaussian_2.run
PASS: beam_events_1.run
PASS: beam_events_2.run
PASS: beam_events_3.run
PASS: beam_events_4.run
PASS: energy_scan_1.run
PASS: restrictions.run
PASS: process_log.run
PASS: shower_err_1.run
PASS: parton_shower_1.run
PASS: parton_shower_2.run
PASS: mlm_matching_fsr.run
XFAIL: user_cuts.run
XFAIL: user_strfun.run
PASS: hepmc_1.run
PASS: hepmc_2.run
PASS: hepmc_3.run
PASS: hepmc_4.run
PASS: hepmc_5.run
PASS: hepmc_6.run
PASS: hepmc_7.run
PASS: hepmc_8.run
PASS: hepmc_9.run
PASS: hepmc_10.run
PASS: analyze_4.run
SKIP: lhapdf5.run
PASS: lhapdf6.run
PASS: stdhep_1.run
PASS: stdhep_2.run
PASS: stdhep_3.run
PASS: stdhep_4.run
PASS: stdhep_5.run
PASS: pythia6_1.run
PASS: pythia6_2.run
PASS: pythia6_3.run
PASS: pythia6_4.run
PASS: mlm_matching_isr.run
PASS: mlm_pythia6_isr.run
PASS: analyze_3.run
PASS: static_1.run
============================================================
Testsuite summary for WHIZARD 2.2.7
============================================================
# TOTAL: 286
# PASS:  281
# SKIP:  2
# XFAIL: 3
# FAIL:  0
# XPASS: 0
# ERROR: 0
============================================================
```

General structure of SINDARIN input

```
model = NMSSM

alias ll = "e-":"e+":"mu+":"mu-"
alias parton = u:U:d:D:s:S:g
alias jet = parton
alias stop = st1:st2:ST1:ST2

process susyprod = parton, parton =>
    stop,stop + gg,gg + gg,stop



sqrts = 13000 GeV
beams = p, p => lhapdf

integrate (susyprod)
     { iterations = 15:500000, 5:1000000 }

n_events = 10000


sample_format = lhef, stdhep, hepmc
sample = "susydata"

simulate (susyprod)
```

LCWS '14, Belgrade, Simulation summary talk:

WHIZARD Task to implement LCIO format

General structure of SINDARIN input

```
model = NMSSM

alias ll = "e-":"e+":"mu+":"mu-"
alias parton = u:U:d:D:s:S:g
alias jet = parton
alias stop = st1:st2:ST1:ST2

process susyprod = parton, parton =>
    stop,stop + gg,gg + gg,stop



sqrts = 13000 GeV
beams = p, p => lhapdf

integrate (susyprod)
     { iterations = 15:500000, 5:1000000 }

n_events = 10000


sample_format = lhef, stdhep, hepmc
sample = "susydata"

simulate (susyprod)
```

LCWS '14, Belgrade, Simulation summary talk:

WHIZARD Task to implement LCIO format

WHIZARD v2.2.4, 02/2015:

```
sample_format = lcio

simulate (<process>)
```

General structure of SINDARIN input

```
model = NMSSM

alias ll = “e-“:”e+”:”mu+”:”mu-“
alias parton = u:U:d:D:s:S:g
alias jet = parton
alias stop = st1:st2:ST1:ST2

process susyprod = parton, parton =>
    stop,stop + gg,gg + gg,stop
```

LCWS ’14, Belgrade, Simulation summary talk:

WHIZARD Task to implement LCIO format

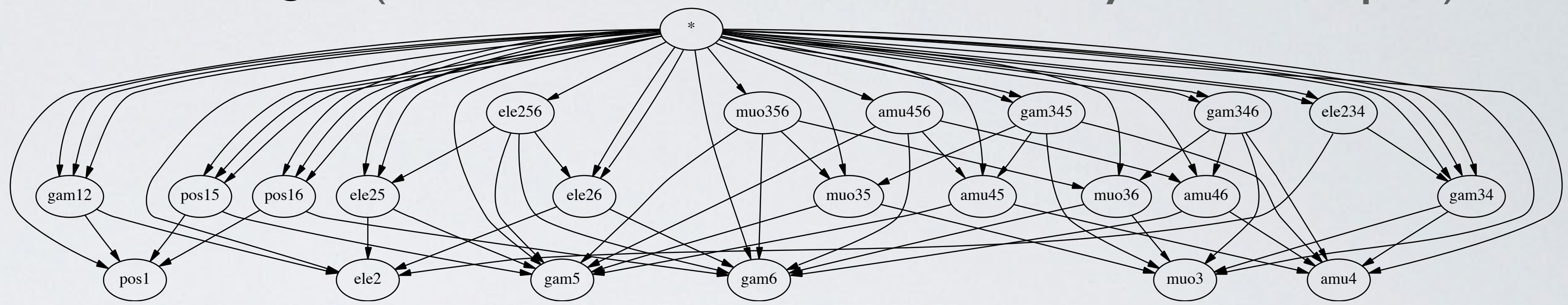
WHIZARD v2.2.4, 02/2015:

```
sample_format = lcio

simulate (<process>)
```

```
===============================================
 - Event  : 1
 - run:  42
 - timestamp 1429387390000000000
 - weight 1
===============================================
 date:      18.04.2015  20:03:10.000000000
 detector : unknown
 event parameters:
 parameter ProcessID [int]: 20,

 collection name : MCParticle
 parameters:

--------------- print out of MCParticle collection ---------------

  flag:  0x0
  simulator status bits: [sbvtcls]  s: created in simulation b: backscatter v: vertex is not endpoint of parent t: decayed in tracker c: decayed in calorimeter l: has left detector s: stopped o: overlay

[   id   ]index|      PDG |    px,       py,        pz      | energy  |gen|[simstat ]| vertex x,      y   ,   z      | endpoint x,    y  ,   z      |    mass |  charge |            spin              | colorflow | [parents] - [daughters]

[00000004]    0|     2212| 0.00e+00, 0.00e+00, 7.00e+03| 7.00e+03| 3 |[s        ]| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (0, 0)   | [] - [2,3]
[00000005]    1|     2212| 0.00e+00, 0.00e+00,-7.00e+03| 7.00e+03| 3 |[s        ]| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (0, 0)   | [] - [2,3]
[00000006]    2|        1| 7.50e-01,-1.57e+00, 3.22e+01| 3.22e+01| 3 |[s        ]| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00| 6.25e-02| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (501, 0)   | [0,1] - [4,5]
[00000007]    3|       -2|-3.05e+00,-1.90e+01,-5.46e+01| 5.79e+01| 3 |[s        ]| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00| 3.38e-01| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (0, 501)   | [0,1] - [4,5]
[00000009]    4|      -24| 1.52e+00,-2.07e+01,-2.06e+01| 8.59e+01| 3 |[s        ]| 0.00e+00, 0.00e+00, 0.00e+00|-3.00e-01, 5.00e-02, 4.00e-03| 8.08e+01| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (0, 0)   | [2,3] - [6,7]
[00000008]    5|       22|-3.81e+00, 1.13e-01,-1.83e+00| 4.23e+00| 1 |[s        ]| 0.00e+00, 0.00e+00, 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00| 8.16e-02| 0.00e+00| 6.00e-01, 1.00e+00, 5.00e-01|  (0, 0)   | [2,3] - []
[00000010]    6|        1|-2.44e+00, 2.88e+01, 6.08e+00| 2.96e+01| 1 |[s        ]|-3.00e-01, 5.00e-02, 4.00e-03| 0.00e+00, 0.00e+00, 0.00e+00|-9.95e-02| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (0, 0)   | [4] - []
[00000011]    7|       -2| 3.96e+00,-4.95e+01,-2.67e+01| 5.64e+01| 1 |[s        ]|-3.00e-01, 5.00e-02, 4.00e-03| 0.00e+00, 0.00e+00, 0.00e+00|-1.74e-01| 0.00e+00| 0.00e+00, 0.00e+00, 0.00e+00|  (0, 0)   | [4] - []

-------------------------------------------------------------------------------
```

Beams, Fields, Colors, Lorentz structures (I)

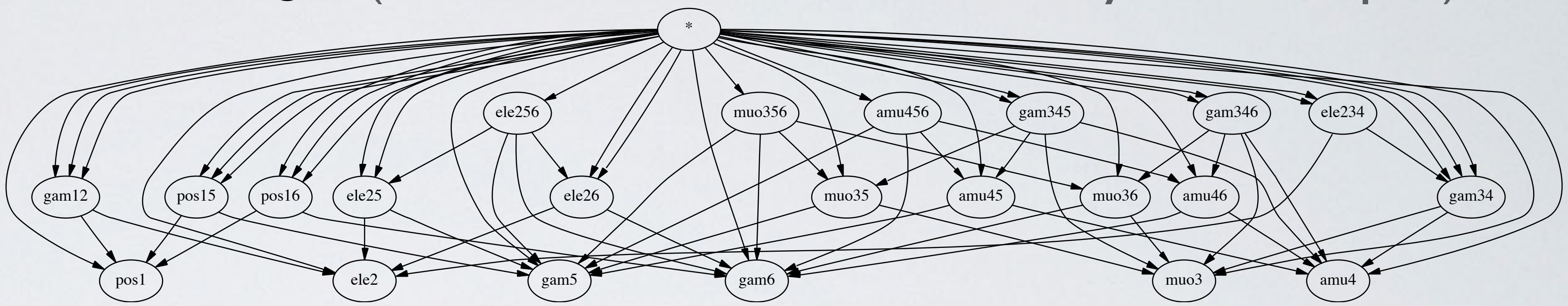
- ✴ Built-in matrix element generator `O'Mega` (recursiveness via Directed Acyclical Graphs)



- ✴ New concept for internal quantum number representation: faster flavor sums, counting of coupling constants (via partial expansion), more speed-up, general Lorentz structures (in prep.)

Beams, Fields, Colors, Lorentz structures (I)

- Built-in matrix element generator `O'Mega` (recursiveness via Directed Acyclical Graphs)



- New concept for internal quantum number representation: faster flavor sums, counting of coupling constants (via partial expansion), more speed-up, general Lorentz structures (in prep.)

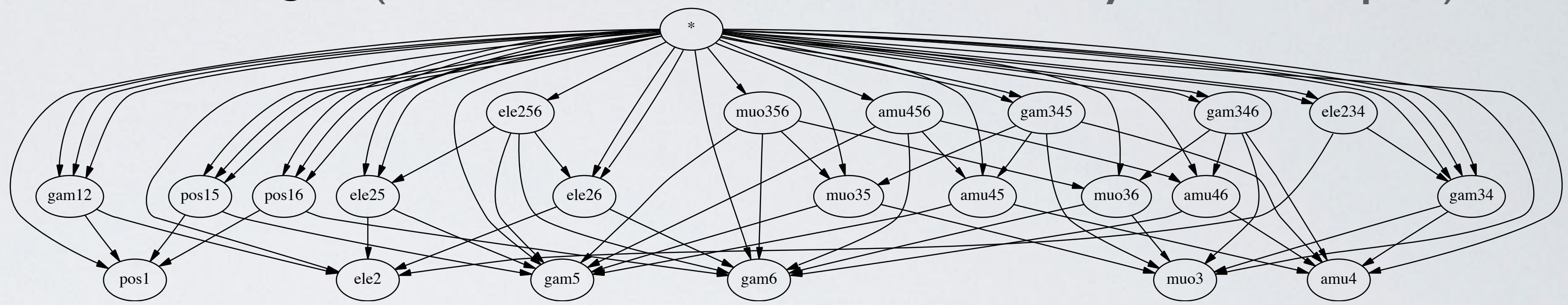
- `O'Mega Virtual Machine (OVM):` matrix elements not as compiled code, but bytecode instructions:

```
process <proc> = in1, in2 => <out>  { $method = "ovm" }
```

```
Flavor states table
2 -2 11 -11
Color flows table: [ (i, j) (k, l) -> (m, n) ...]
1 0 0 -1 0 0 0 0
Color ghost flags table:
0 0 0 0
Color factors table: [ i, j: num den power], i, j are indexed color flows
        1        1        1        1        1
Flavor color combination is allowed:
1
OVM instructions for momenta addition, fusions and brakets start here:
0 0 0 0 0 0 0 0
1 0 0 5 1 2 0 0
11 2 0 2 1 0 0 1
13 2 0 2 2 0 0 1
14 11 0 1 3 0 0 1
12 11 0 1 4 0 0 1
0 0 0 0 0 0 0 0
60 22 2 1 5 0 0 1
-1 2 1 1 2 2 0 0
58 23 2 2 5 0 0 1
-4 3 1 2 2 2 0 0
0 0 0 0 0 0 0 0
2 -1 0 1 1 0 0 0
-4 2 1 2 1 1 0 0
-1 1 1 1 1 1 0 0
```

Beams, Fields, Colors, Lorentz structures (I)

✳ Built-in matrix element generator `O'Mega` (recursiveness via Directed Acyclical Graphs)



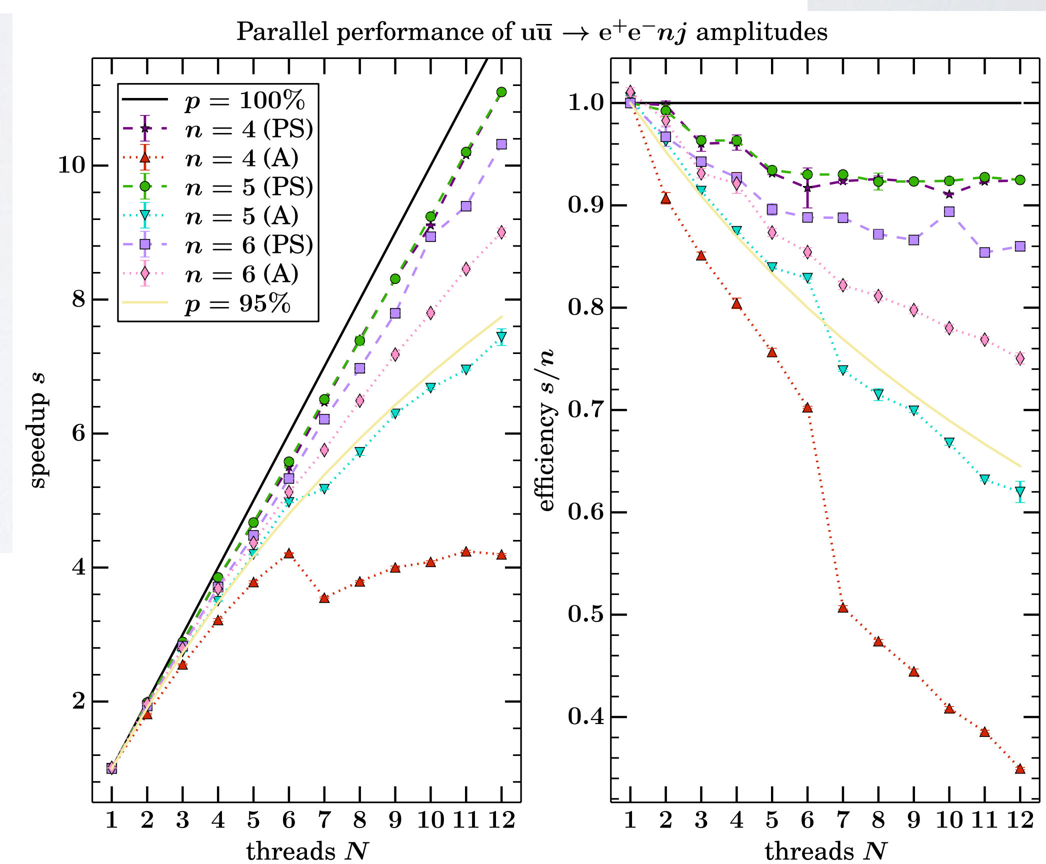
✳ New concept for internal quantum number representation: faster flavor sums, counting of coupling constants (via partial expansion), more speed-up, general Lorentz structures (in prep.)

✳ `O'Mega Virtual Machine (OVM):` matrix elements not as compiled code, but bytecode instructions:

```
process <proc> = in1, in2 => <out>  { $method = "ovm" }
```

```
Flavor states table
2 -2 11 -11
Color flows table: [ (i, j) (k, l) -> (m, n) ...]
1 0 0 -1 0 0 0 0
Color ghost flags table:
0 0 0 0
Color factors table: [ i, j: num den power], i, j are indexed color flows
        1        1        1        1        1
Flavor color combination is allowed:
1
OVM instructions for momenta addition, fusions and brakets start here:
0 0 0 0 0 0 0 0
1 0 0 5 1 2 0 0
11 2 0 2 1 0 0 1
13 2 0 2 2 0 0 1
14 11 0 1 3 0 0 1
12 11 0 1 4 0 0 1
0 0 0 0 0 0 0 0
60 22 2 1 5 0 0 1
-1 2 1 1 2 2 0 0
58 23 2 2 5 0 0 1
-4 3 1 2 2 2 0 0
0 0 0 0 0 0 0 0
2 -1 0 1 1 0 0 0
-4 2 1 2 1 1 0 0
-1 1 1 1 1 1 0 0
```

process	BC size	Fortran size	t_{compile}
$gg \to gggggg$	428 MiB	4.0 GiB	-
$gg \to ggggg$	9.4 MiB	85 MiB	483(18) s
$gg \to q\bar{q}q'\bar{q}'q''\bar{q}''g$	3.2 MiB	27 MiB	166(15) s
$e^+e^- \to 5\,(e^+e^-)$	0.7 MiB	1.9 MiB	32.46(13) s



Beams, Fields, Colors, Lorentz structures (II)

Particle types:

- Spin 0 particles
- Spin 1/2 particles (Dirac and Majorana, Fermi statistics for both fermion-number conserving and violating Feynman rules)
- Spin 1 particles (massive+massless, unitarity/Feynman/R_ξ gauges)
- Spin 3/2 particles (Majorana only, gravitinos)
- Spin 2 particles (massive+massless, more about tensors later)
- Dynamic particles and also pure insertions
- Unphysical particles [ghosts] for Ward- and Slavnov-Taylor identities

Beams, Fields, Colors, Lorentz structures (II)

Particle types:

- Spin 0 particles
- Spin 1/2 particles (Dirac and Majorana, Fermi statistics for both fermion-number conserving and violating Feynman rules)
- Spin 1 particles (massive+massless, unitarity/Feynman/R_ξ gauges)
- Spin 3/2 particles (Majorana only, gravitinos)
- Spin 2 particles (massive+massless, more about tensors later)
- Dynamic particles and also pure insertions
- Unphysical particles [ghosts] for Ward- and Slavnov-Taylor identities

Gravitinos,
JRR 2001

```
*** Checking polarization vectorspinors: ***
p.ueps ( 2)= 0: passed at   86%
p.ueps ( 1)= 0: passed at   86%
.....
*** Checking the irreducibility condition: ***
g.ueps ( 2): passed at   95%
.....
g.ueps (-2): passed at   95%
g.veps ( 2): passed at   95%
.....
g.veps (-2): passed at   95%
*** Testing vectorspinor normalization ***
ueps( 2).ueps( 2)= -2m: passed at  100%
ueps( 1).ueps( 1)= -2m: passed at  100%
.....
*** Majorana properties of gravitino vertices: ***
f_sgr     + gr_sf     = 0: passed at   84%
slr_grf   + slr_fgr   = 0: passed at   88%
.....
v2lr_fgr  + v2lr_grf  = 0: passed at   77% [expected  0.000E+00, got  0.633E-12]
*** Testing the gravitino propagator: ***
Transversality:
p.pr.test: passed at   66% [expected  0.000E+00, got  0.437E-10]
p.pr.ueps ( 2): passed at   86%
.....
p.pr.veps (-2): passed at   79% [expected  0.000E+00, got  0.342E-12]
Irreducibility:
g.pr.test: passed at   78% [expected  0.000E+00, got  0.471E-12]
g.pr.ueps ( 2): passed at   92%
```

Beams, Fields, Colors, Lorentz structures (II)

Particle types:

- Spin 0 particles
- Spin 1/2 particles (Dirac and Majorana, Fermi statistics for both fermion-number conserving and violating Feynman rules)
- Spin 1 particles (massive+massless, unitarity/Feynman/R_ξ gauges)
- Spin 3/2 particles (Majorana only, gravitinos)
- Spin 2 particles (massive+massless, more about tensors later)
- Dynamic particles and also pure insertions
- Unphysical particles [ghosts] for Ward- and Slavnov-Taylor identities

Gravitinos, JRR 2001

```
*** Checking polarization vectorspinors: ***
p.ueps ( 2)= 0: passed at   86%
p.ueps ( 1)= 0: passed at   86%
.....
*** Checking the irreducibility condition: ***
g.ueps ( 2): passed at   95%
.....
g.ueps (-2): passed at   95%
g.veps ( 2): passed at   95%
.....
g.veps (-2): passed at   95%
*** Testing vectorspinor normalization ***
ueps( 2).ueps( 2)= -2m: passed at  100%
ueps( 1).ueps( 1)= -2m: passed at  100%
.....
*** Majorana properties of gravitino vertices: ***
f_sgr    + gr_sf    = 0: passed at   84%
slr_grf  + slr_fgr  = 0: passed at   88%
.....
v2lr_fgr  + v2lr_grf  = 0: passed at   77% [expected  0.000E+00, got  0.633E-1
*** Testing the gravitino propagator: ***
Transversality:
p.pr.test: passed at   66% [expected  0.000E+00, got  0.437E-10]
p.pr.ueps ( 2): passed at   86%
.....
p.pr.veps (-2): passed at   79% [expected  0.000E+00, got  0.342E-12]
Irreducibility:
g.pr.test: passed at   78% [expected  0.000E+00, got  0.471E-12]
g.pr.ueps ( 2): passed at   92%
```

Gravitons, Ohl 2000

```
*** Checking polarisation tensors: ***
e2( 2).e2( 2)=1: passed at  100%
e2( 2).e2(-2)=0: passed at  100%
....
e2( 0).e2( 2)=0: passed at  100%
e2( 0).e2( 1)=0: passed at   94%
....
 |p.e2( 2)|   =0: passed at   96%
   |e2( 2).p|=0: passed at   96%
 |p.e2(-2)|   =0: passed at   96%
   |e2(-2).p|=0: passed at   96%
 |p.e2( 1)|   =0: passed at   88%
   |e2( 1).p|=0: passed at   88%
 |p.e2( 0)|   =0: passed at   84%
   |e2( 0).p|=0: passed at   84%
 |p.e2(-1)|   =0: passed at   88%
   |e2(-1).p|=0: passed at   88%
 *** Checking the graviton propagator:
p.pr.e(-2): passed at   90%
p.pr.e(-1): passed at   82%
p.pr.e(0): passed at   82%
p.pr.e(1): passed at   82%
p.pr.e(2): passed at   90%
p.pr.ttest: passed at   74% [expected  0.000E+00, got  0
```

Beams, Fields, Colors, Lorentz structures (III)

Lorentz structures:

- Large number of hardcoded terms: pure scalar, pure vector, scalar/vector, fermion/scalar, fermion/vector, fermion/tensor, vector/tensor, gravitino couplings, fermion coupl. SUSY Ward id.
- Growing number of dim. 5/6/7/8 operators: HEFT, aTGCs, aQGCs, anomalous top couplings etc.
- Completely general Lorentz structures: foreseen for major next release (incl. UFO support), v2.3.0

Beams, Fields, Colors, Lorentz structures (III)

Lorentz structures:

- Large number of hardcoded terms: pure scalar, pure vector, scalar/vector, fermion/scalar, fermion/vector, fermion/tensor, vector/tensor, gravitino couplings, fermion coupl. SUSY Ward id.
- Growing number of dim. 5/6/7/8 operators: HEFT, aTGCs, aQGCs, anomalous top couplings etc.
- Completely general Lorentz structures: foreseen for major next release (incl. UFO support), v2.3.0

Color structures:

- Color flow formalism Stelzer/Willenbrock, 2003; Kilian/Ohl/JRR/Speckner, 2011
- Fundamental, antifundamental and adjoint representations
- Inofficial version for color sextets and diquark couplings
- General color structures coming tied to general Lorentz structures

Beams, Fields, Colors, Lorentz structures (III)

Lorentz structures:

- Large number of hardcoded terms: pure scalar, pure vector, scalar/vector, fermion/scalar, fermion/vector, fermion/tensor, vector/tensor, gravitino couplings, fermion coupl. SUSY Ward id.
- Growing number of dim. 5/6/7/8 operators: HEFT, aTGCs, aQGCs, anomalous top couplings etc.
- Completely general Lorentz structures: foreseen for major next release (incl. UFO support), v2.3.0

Color structures:

- Color flow formalism Stelzer/Willenbrock, 2003; Kilian/Ohl/JRR/Speckner, 2011
- Fundamental, antifundamental and adjoint representations
- Inofficial version for color sextets and diquark couplings
- General color structures coming tied to general Lorentz structures

Beams:

- Lepton beam ISR Kuraev/Fadin, 2003; Skrzypek/Jadach, 1991
- Lepton collider beams: `CIRCE1/2`, also photon beams (more later)
- PDFs: interface to `LHAPDF v4/5/6`; internal PDFs: CTEQ6, CT10, MMHT etc.
- QCD parton shower: 2 own implementations [or ext., more later]

BSM Models in WHIZARD

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 for e^+e^- top threshold	—	SM_tt_threshold
SM with anom. Higgs coupl.	—	SM_rx / NoH
SM ext. for VV scattering	—	SSC / SSC2/ AltH
SM ext. for unitarity limits	—	SM_ul
SM with Z'	—	Zprime
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/univ.)	—	Simplest[_univ]
3-site model	—	Threeshl
UED	—	UED
SM with gravitino and photino	—	GravTest
Augmentable SM template	—	Template

BSM Models in WHIZARD

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 for e^+e^- top threshold	—	SM_tt_threshold
SM with anom. Higgs coupl.	—	SM_rx / NoH
SM ext. for VV scattering	—	SSC / SSC2/ AltH
SM ext. for unitarity limits	—	SM_ul
SM with Z'	—	Zprime
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/univ.)	—	Simplest[_univ]
3-site model	—	Threeshl
UED	—	UED
SM with gravitino and photino	—	GravTest
Augmentable SM template	—	Template

- Automated models: interface to SARAH/BSM Toolbox Staub, 0909.2863; Ohl/Porod/Staub/Speckner, 1109.5147
- Automated models: interface to FeynRules Christensen/Duhr; Christensen/Duhr/Fuks/JRR/Speckner, 1010.3251

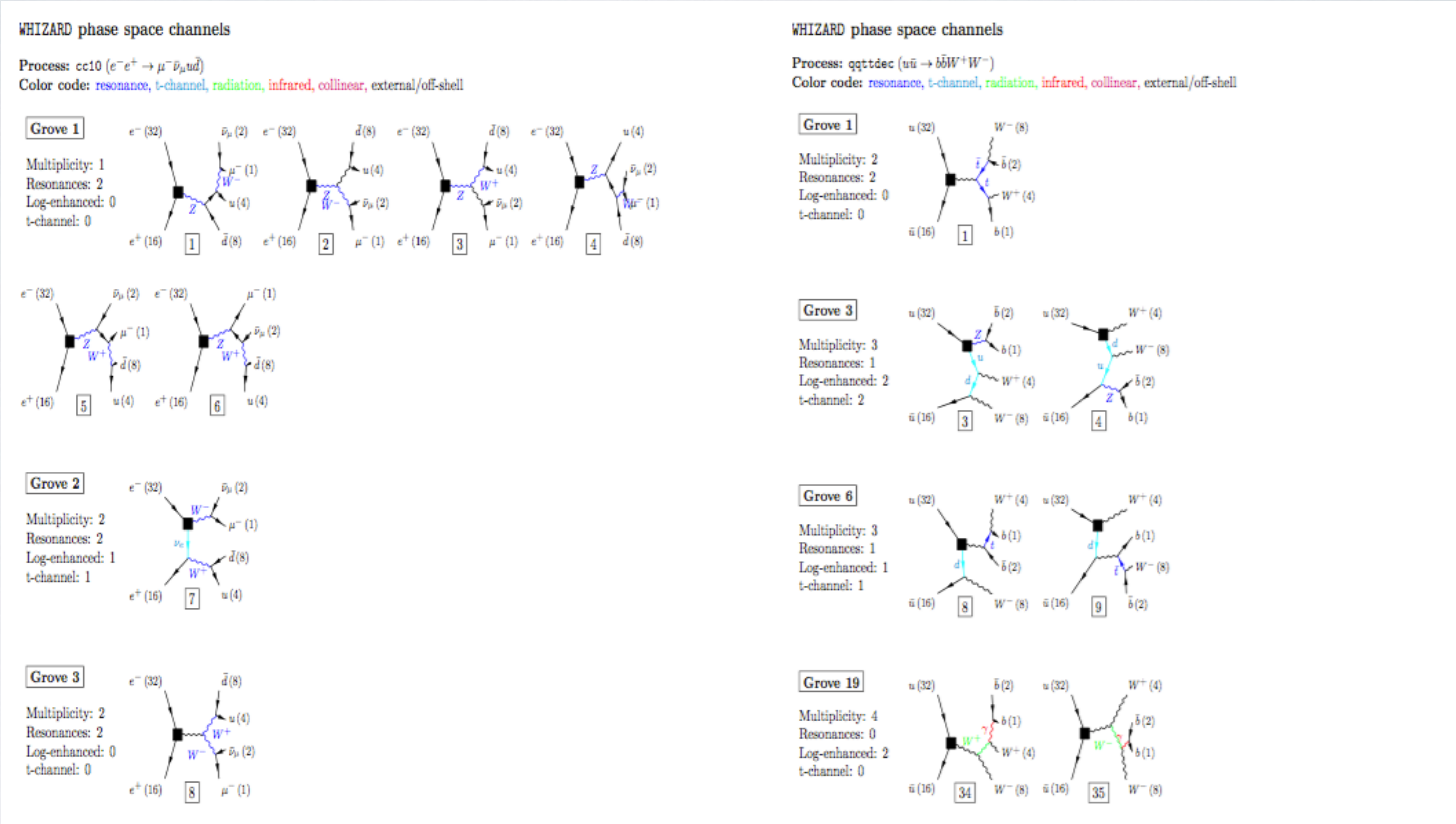
BSM Models in WHIZARD

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 for e^+e^- top threshold	—	SM_tt_threshold
SM with anom. Higgs coupl.	—	SM_rx / NoH
SM ext. for VV scattering	—	SSC / SSC2/ AltH
SM ext. for unitarity limits	—	SM_ul
SM with Z'	—	Zprime
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/univ.)	—	Simplest[_univ]
3-site model	—	Threeshl
UED	—	UED
SM with gravitino and photino	—	GravTest
Augmentable SM template	—	Template

- Automated models: interface to SARAH/BSM Toolbox Staub, 0909.2863; Ohl/Porod/Staub/Speckner, 1109.5147
- Automated models: interface to FeynRules Christensen/Duhr; Christensen/Duhr/Fuks/JRR/Speckner, 1010.3251
- Automated models: UFO interface [in connection with new WHIZARD/O'Mega model format]

Phase Space Setup

WHIZARD algorithm: heuristics to classify phase-space topology, adaptive multi-channel mapping $\Longrightarrow$ `resonant, t-channel, radiation, infrared, collinear, off-shell`



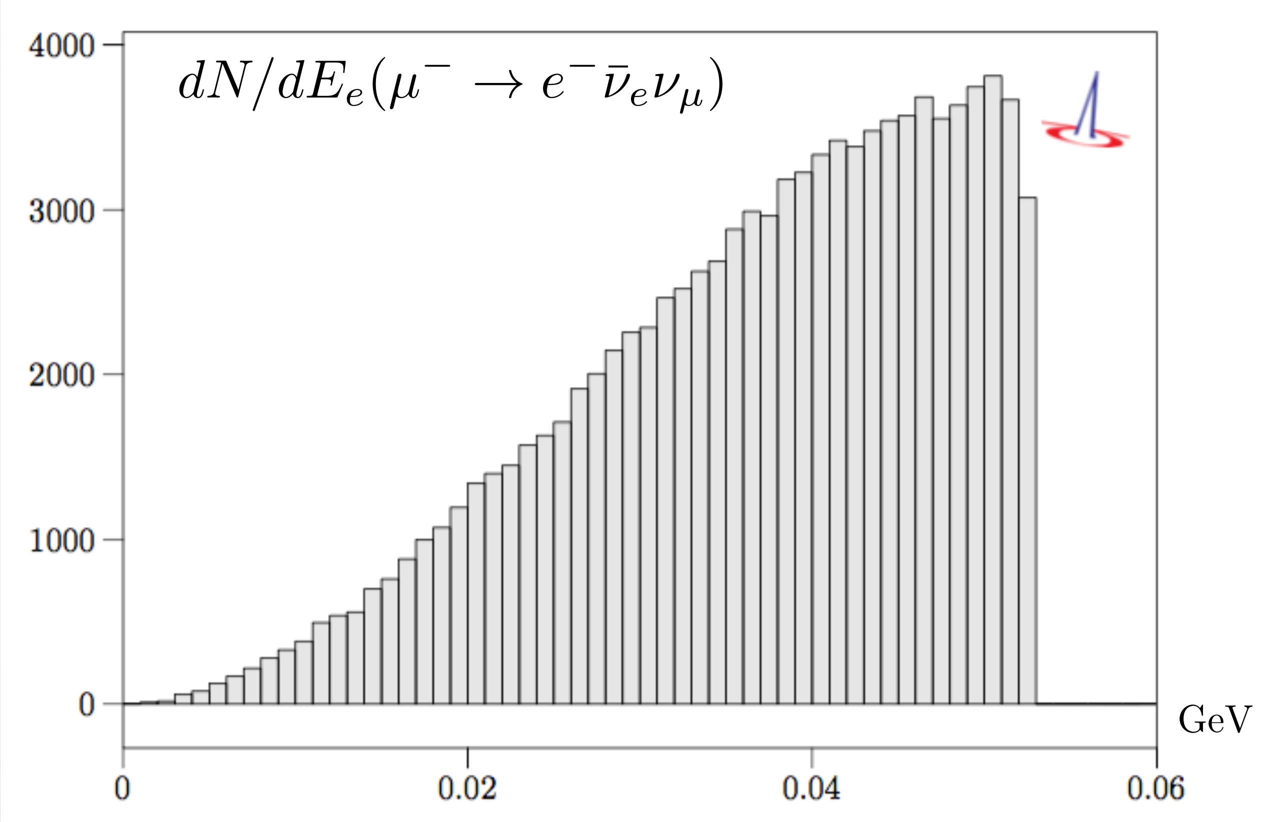
Complicated processes: factorization into production and decay with the `unstable` option

Decay processes / auto_decays

WHIZARD cannot only do scattering processes, but also decays

Example Energy distribution electron in muon decay:

```
model = SM
process mudec = e2 => e1, N1, n2
integrate (mudec)

histogram e_e1 (0, 60 MeV, 1 MeV)
analysis = record e_e1 (eval E [e1])

n_events = 100000

simulate (mudec)

compile_analysis { $out_file = “test.dat” }
```

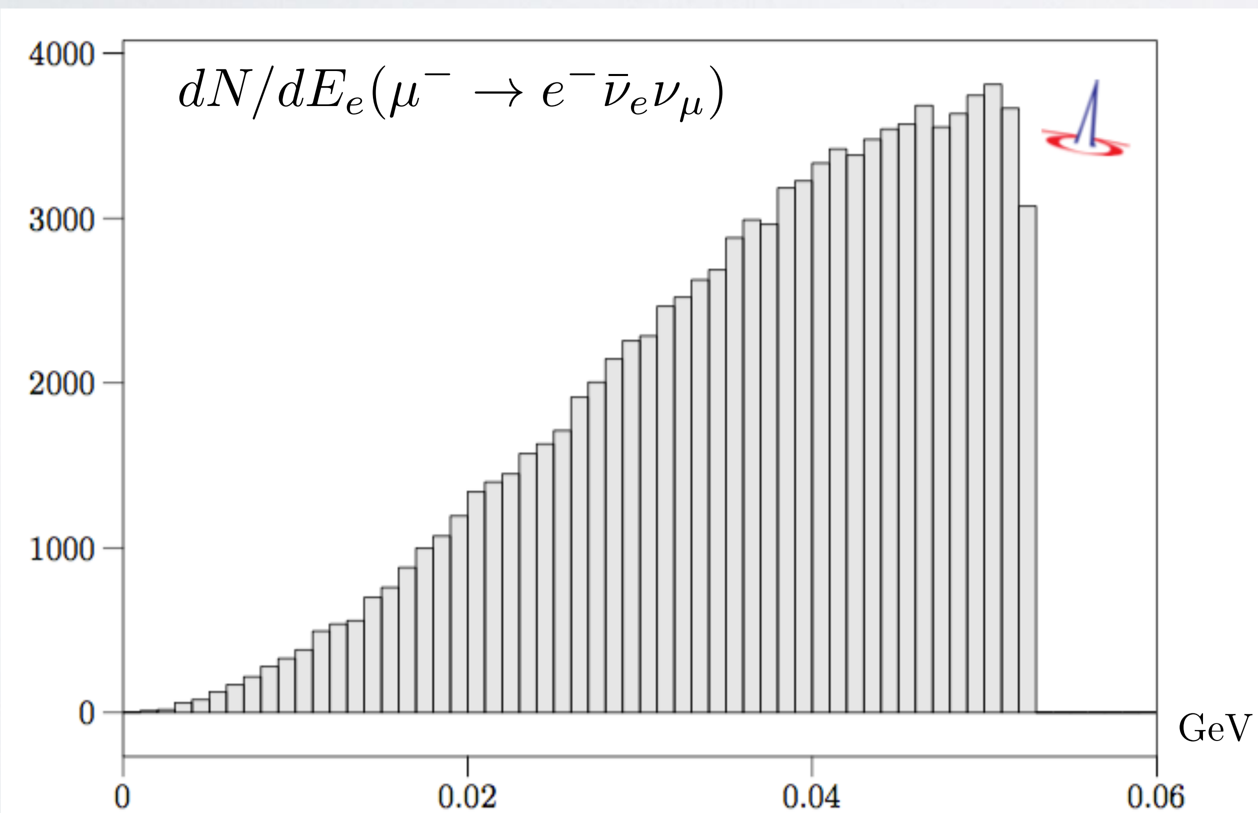


Decay processes / auto_decays

WHIZARD cannot only do scattering processes, but also decays

Example Energy distribution electron in muon decay:

```
model = SM
process mudec = e2 => e1, N1, n2
integrate (mudec)

histogram e_e1 (0, 60 MeV, 1 MeV)
analysis = record e_e1 (eval E [e1])

n_events = 100000

simulate (mudec)

compile_analysis { $out_file = "test.dat" }
```



Automatic integration of particle decays

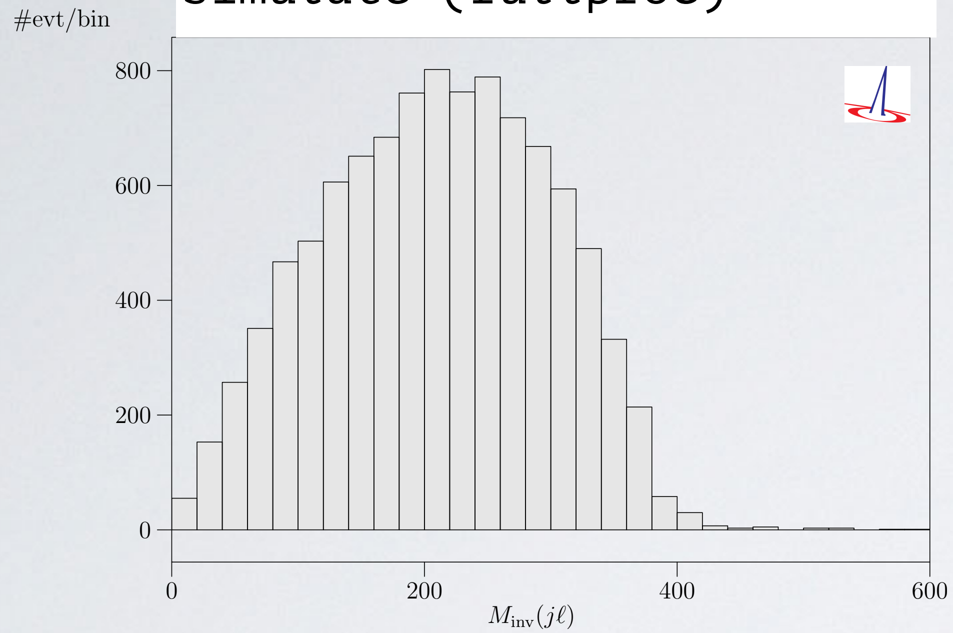
```
auto_decays_multiplicity = 2
?auto_decays_radiative = false

unstable Wp () { ?auto_decays = true }
```

```
|=============================================================
| It      Calls  Integral[GeV]  Error[GeV]   Err[%]    Acc
|=============================================================
   1        100  2.2756406E-01  0.00E+00     0.00     0.00*
|-------------------------------------------------------------
   1        100  2.2756406E-01  0.00E+00     0.00     0.00
|=============================================================
| Unstable particle W+: computed branching ratios:
|   decay_p24_1: 3.3337068E-01   dbar, u
|   decay_p24_2: 3.3325864E-01   sbar, c
|   decay_p24_3: 1.1112356E-01   e+, nue
|   decay_p24_4: 1.1112356E-01   mu+, numu
|   decay_p24_5: 1.1112356E-01   tau+, nutau
|   Total width = 2.0478471E+00 GeV (computed)
|               = 2.0490000E+00 GeV (preset)
|   Decay options: helicity treated exactly
```

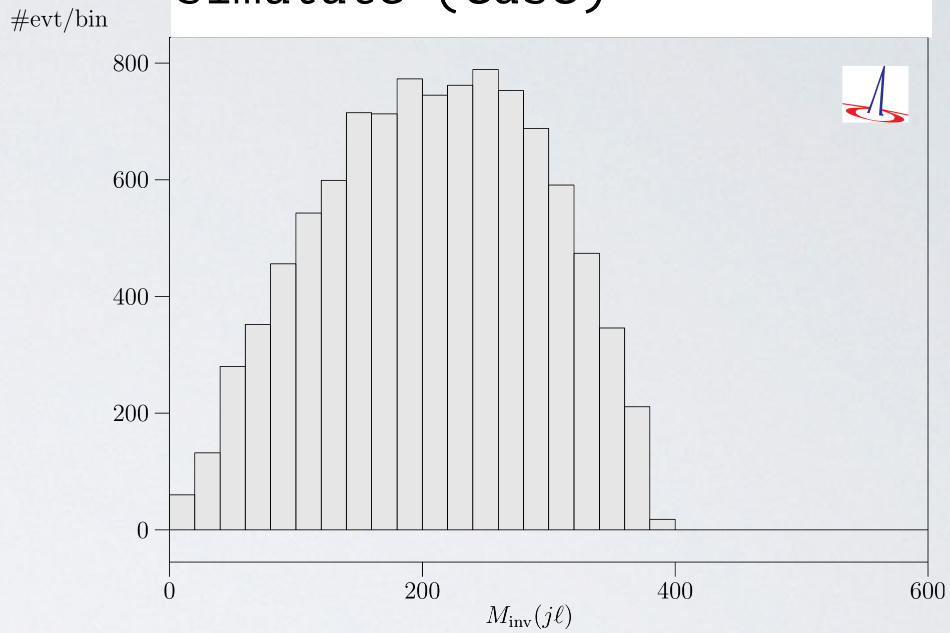
Spin Correlation and Polarization in Cascades

Cascade decay, factorize production and decay

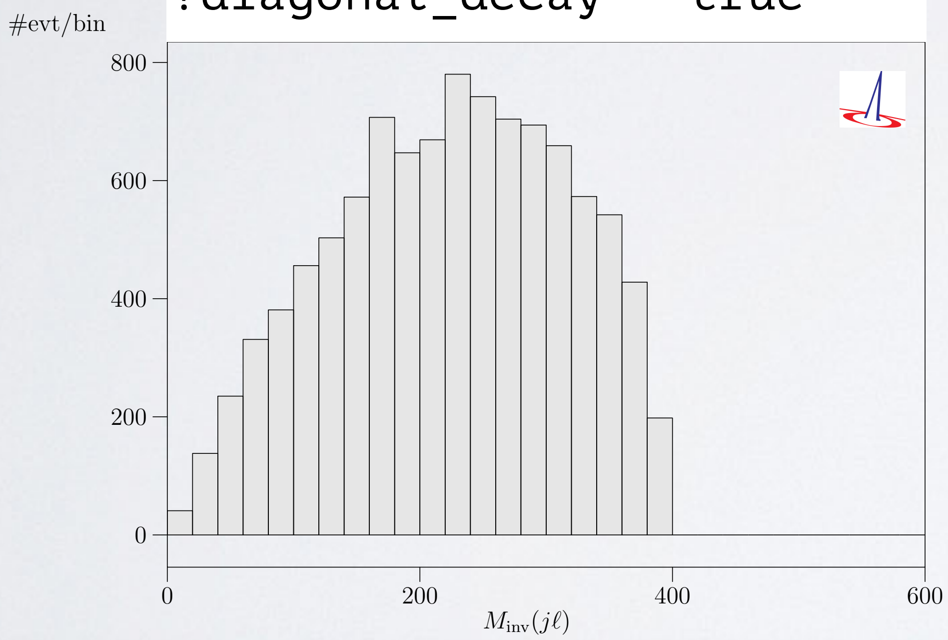
`simulate (fullproc)`



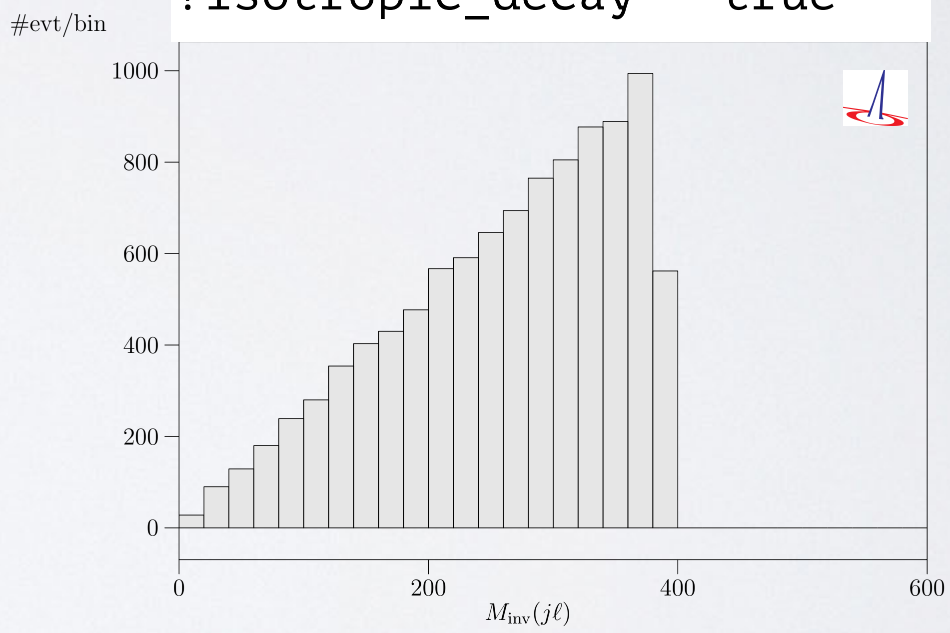
`simulate (casc)`



`?diagonal_decay = true`



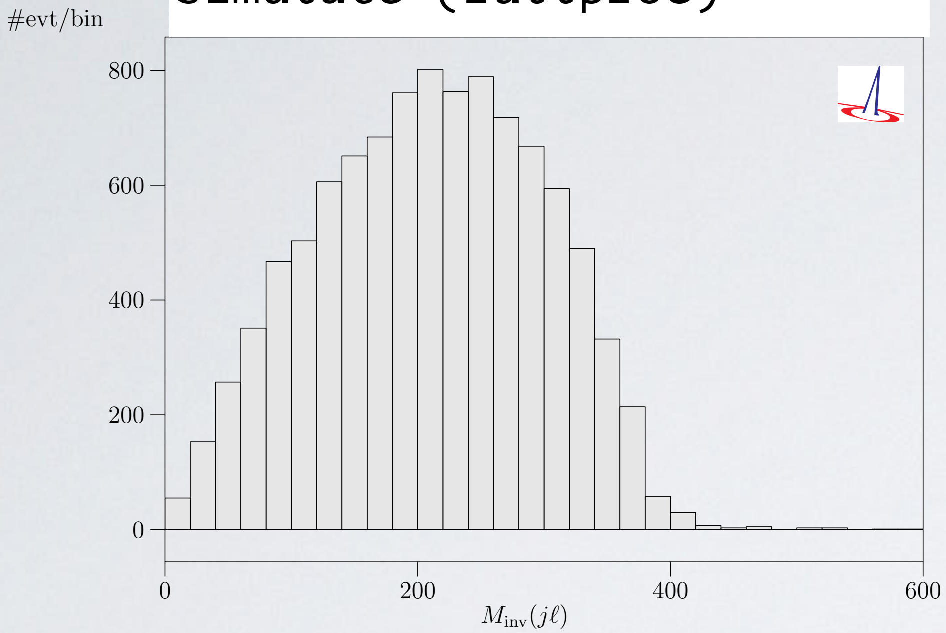
`?isotropic_decay = true`



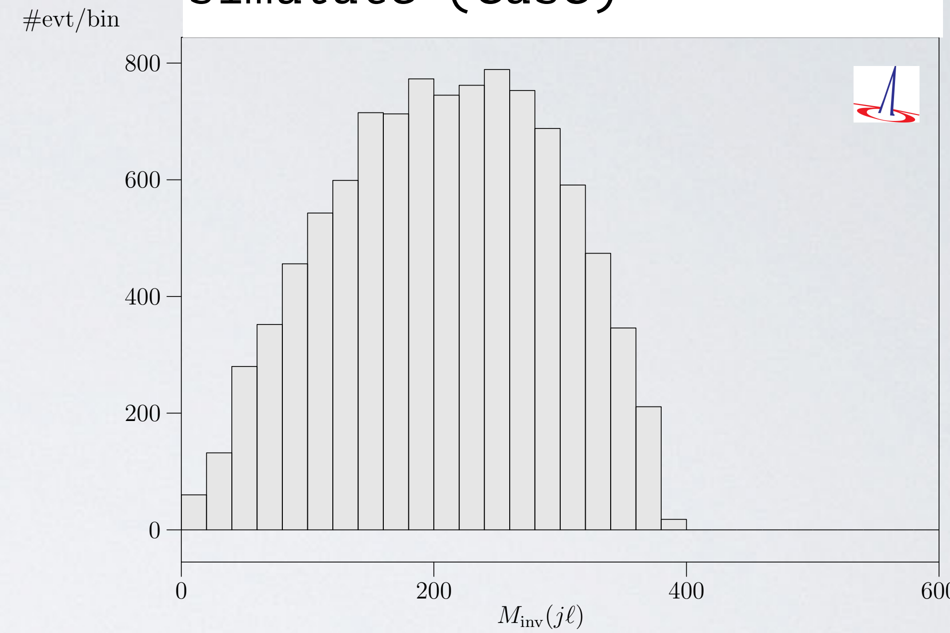
Spin Correlation and Polarization in Cascades

Cascade decay, factorize production and decay

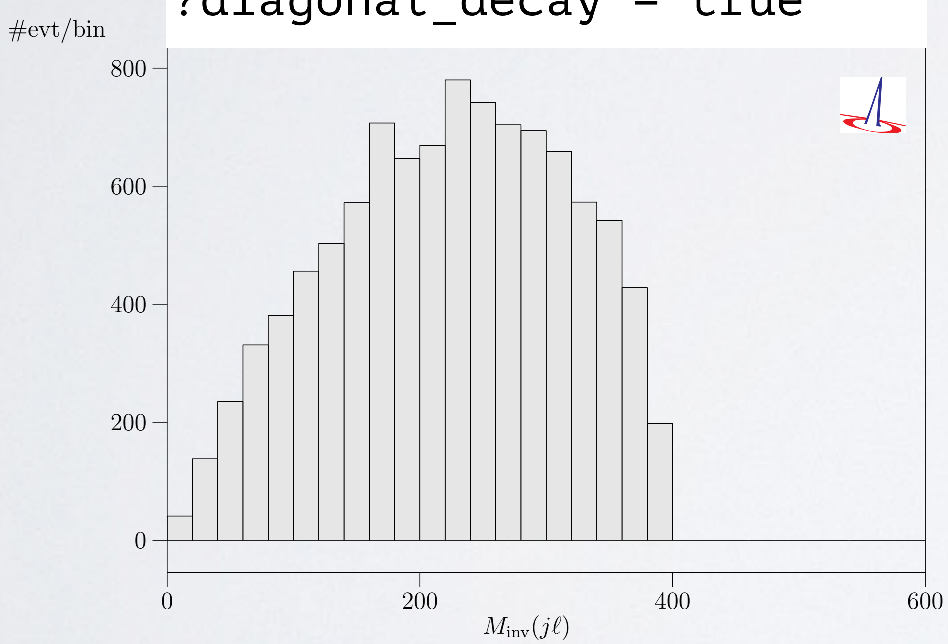
`simulate (fullproc)`



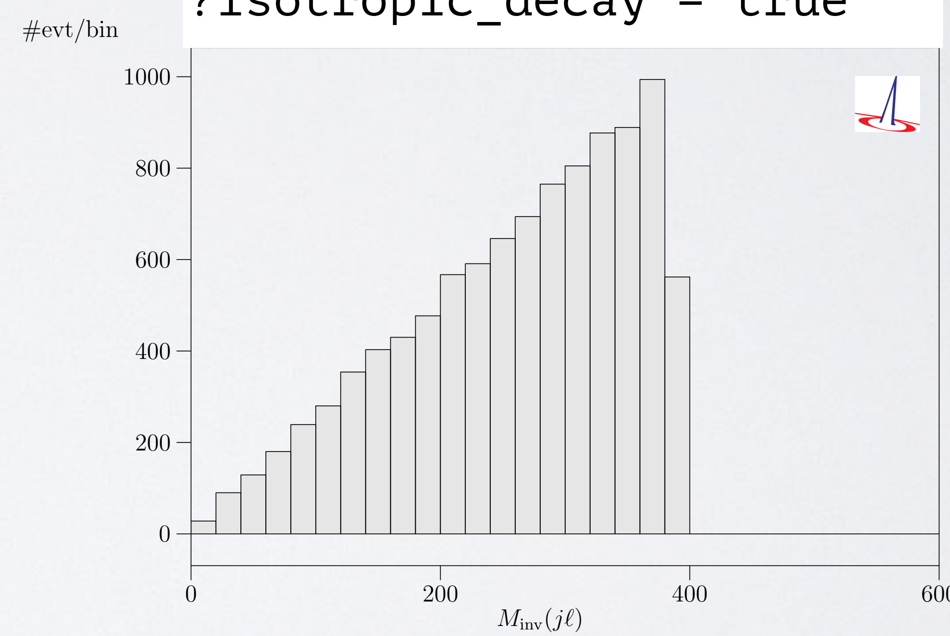
`simulate (casc)`



`?diagonal_decay = true`



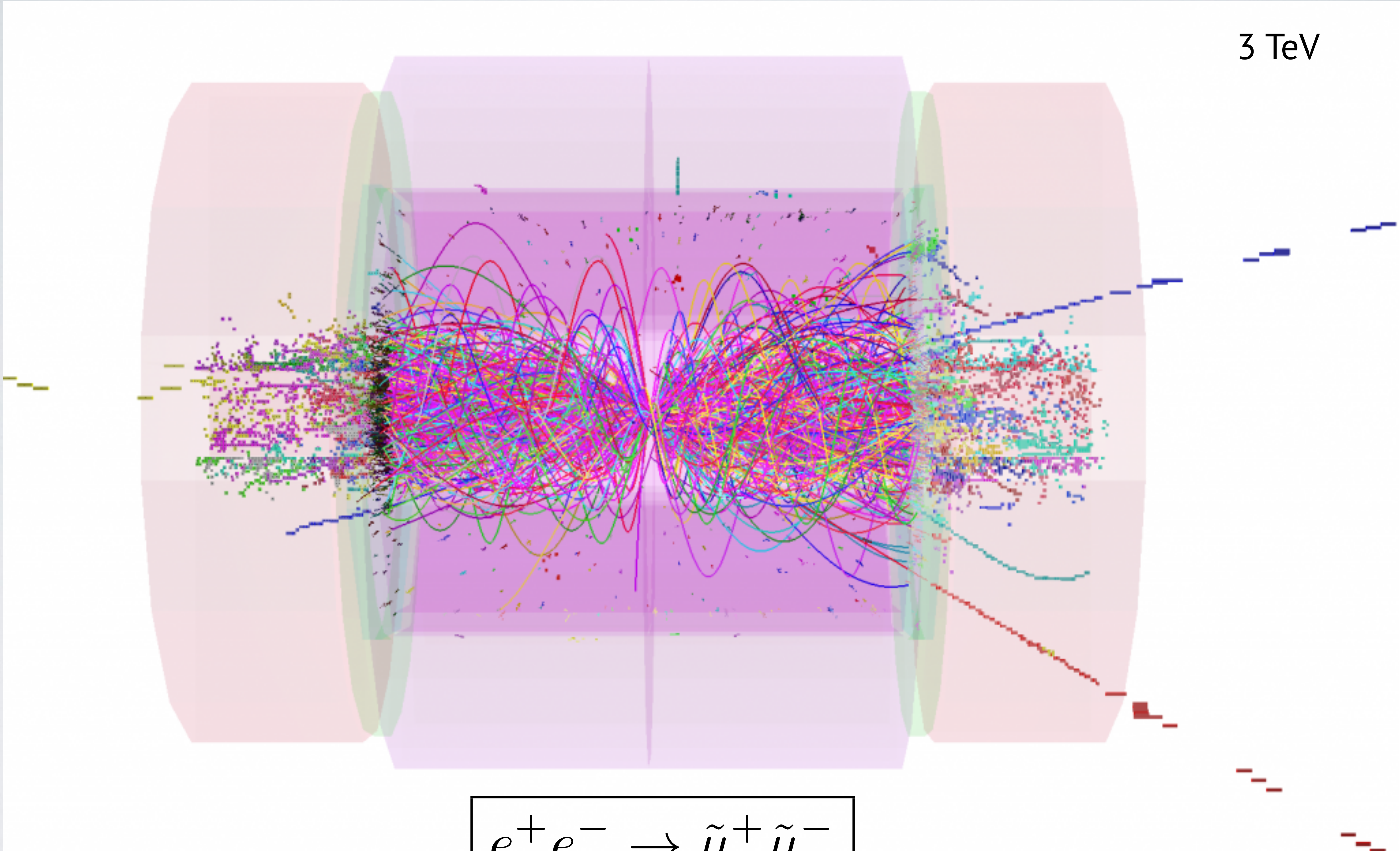
`?isotropic_decay = true`



NEW: possibility to select specific helicity in decays!

```
unstable “W+” { decay_helicity = 0 }
```

Lepton Collider Beam Simulation

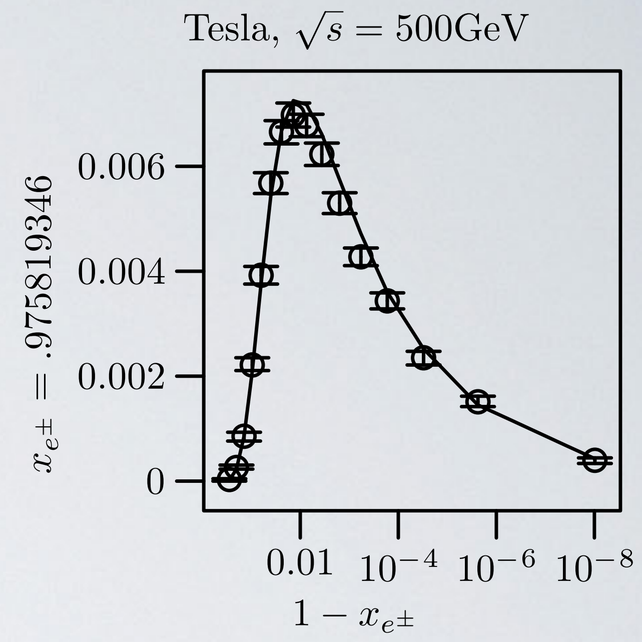


$$e^+e^- \to \tilde{\mu}^+\tilde{\mu}^-$$

Courtesy to Philipp Roloff

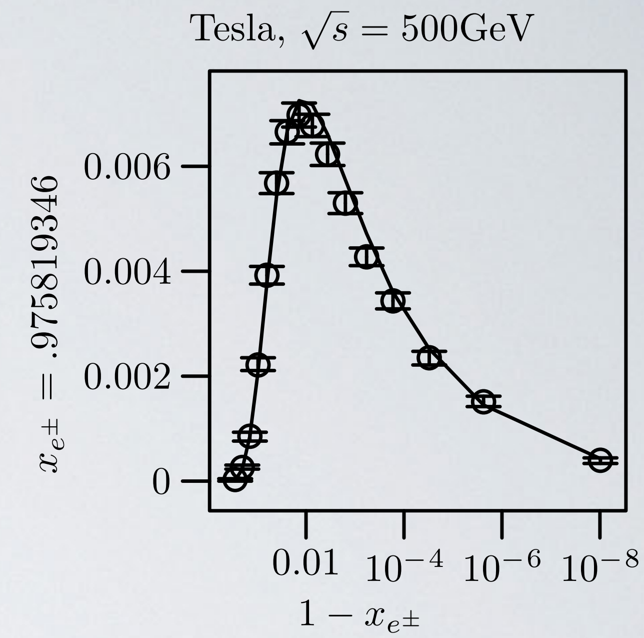
Lepton Collider Beam Simulation

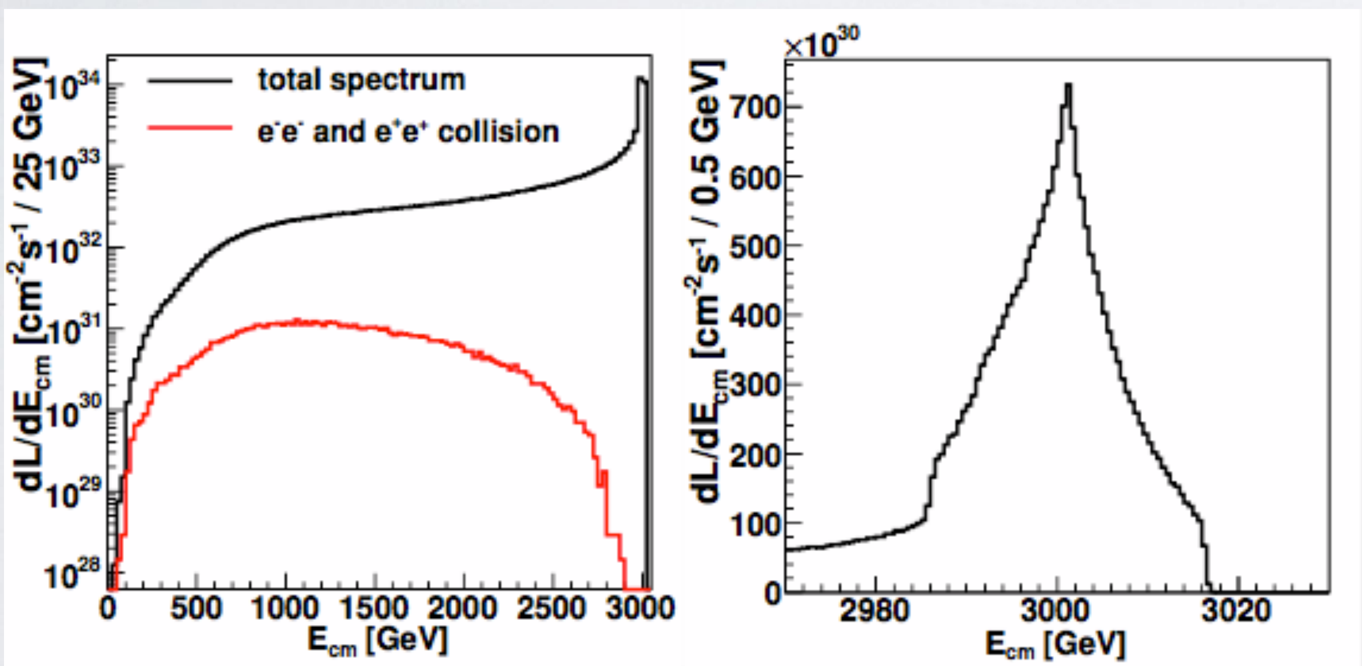
- Another demand: adapt `GuineaPig` beam spectra for `WHIZARD v2`
- For `WHIZARD v1.95` simulations done by `Lumilinker` [T. Barklow]
- TESLA/SLC spectra were rather simple
- Fits with 6 or 7 parameters possible [`CIRCE1`]
- Beams not factorizable: $D_{B_1 B_2}(x_1, x_2) \neq D_{B_1}(x_1) \cdot D_{B_2}(x_2)$
- No simple power law: $D_{B_1 B_2}(x_1, x_2) \neq x_1^{\alpha_1}(1-x_1)^{\beta_1} x_2^{\alpha_2}(1-x_2)^{\beta_2}$



Lepton Collider Beam Simulation

- Another demand: adapt `GuineaPig` beam spectra for `WHIZARD v2`
- For `WHIZARD v1.95` simulations done by `Lumilinker` [T. Barklow]
- TESLA/SLC spectra were rather simple
- Fits with 6 or 7 parameters possible [`CIRCE1`]
- Beams not factorizable: $D_{B_1B_2}(x_1,x_2) \neq D_{B_1}(x_1)\cdot D_{B_2}(x_2)$
- No simple power law: $D_{B_1B_2}(x_1,x_2) \neq x_1^{\alpha_1}(1-x_1)^{\beta_1} x_2^{\alpha_2}(1-x_2)^{\beta_2}$



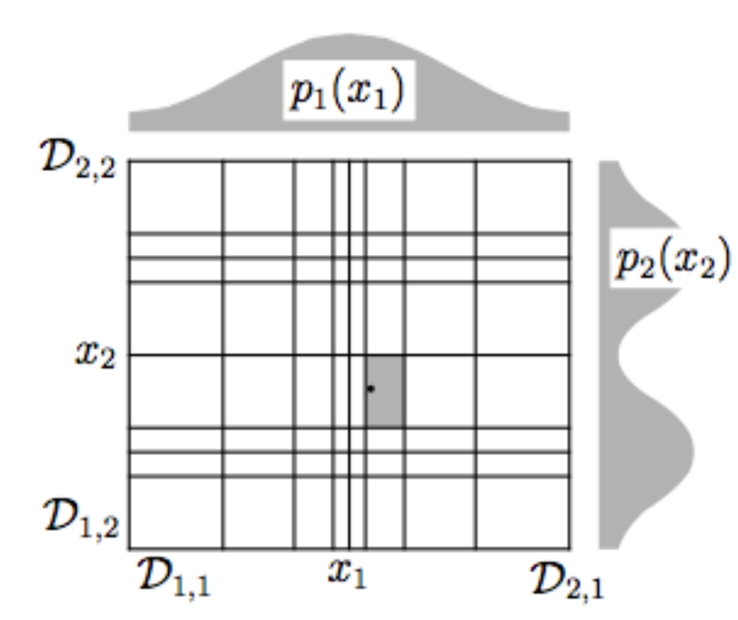


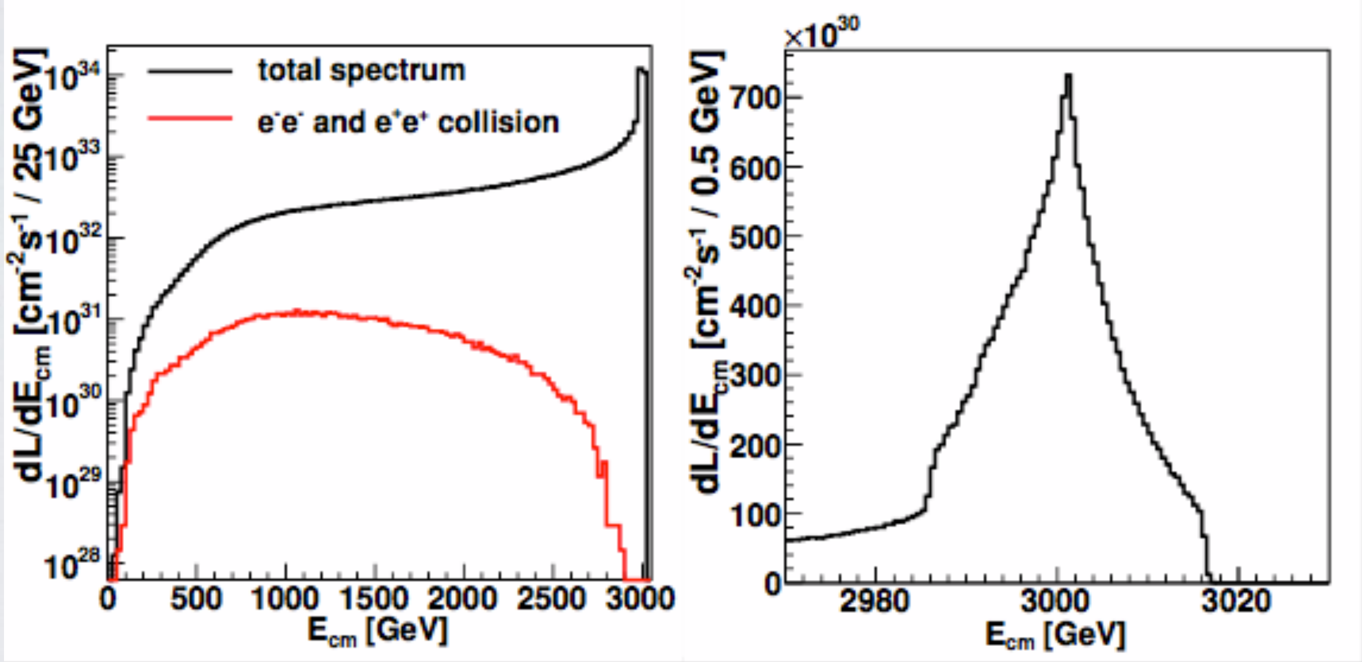
Dalena/Esbjerg/Schulte [LCWS 2011]

Tails @ CLIC much more complicated (wakefields)

Lepton Collider Beam Simulation

- Another demand: adapt `GuineaPig` beam spectra for `WHIZARD v2`
- For `WHIZARD v1.95` simulations done by `Lumilinker` [T. Barklow]
- TESLA/SLC spectra were rather simple
- Fits with 6 or 7 parameters possible [`CIRCE1`]
- Beams not factorizable: $D_{B_1 B_2}(x_1, x_2) \neq D_{B_1}(x_1) \cdot D_{B_2}(x_2)$
- No simple power law: $D_{B_1 B_2}(x_1, x_2) \neq x_1^{\alpha_1}(1-x_1)^{\beta_1} x_2^{\alpha_2}(1-x_2)^{\beta_2}$





Dalena/Esbjerg/Schulte [LCWS 2011]

Tails @ CLIC much more complicated (wakefields)

`CIRCE2` algorithm (`WHIZARD 2.2.5`, 02/15)

- Adapt 2D factorized variable width histogram to steep part of distribution
- Smooth correlated fluctuations with moderate Gaussian filter [suppresses artifacts from limited GuineaPig statistics
- Smooth continuum/boundary bins separately [avoid artificial beam energy spread]

Workflow GuineaPig/CIRCE2/WHIZARD

1. Run Guinea-Pig++ with

```
do_lumi=7;num_lumi=100000000;num_lumi_eg=100000000;num_lumi_gg=100000000;
```

to produce lumi.[eg][eg].out with (E_1, E_2) pairs.

[Large event numbers, as Guinea-Pig++ will produce only a small fraction!]

2. Run circe2_tool.opt with steering file

```
{ file="ilc500/beams.circe"                         # to be loaded by WHIZARD
  { design="ILC" roots=500 bins=100 scale=250 # E in [0,1]
    { pid/1=electron pid/2=positron pol=0         # unpolarized e-/e+
      events="ilc500/lumi.ee.out" columns=2     # <= Guinea-Pig
      lumi = 1564.763360                             # <= Guinea-Pig
      iterations = 10                                  # adapting bins
      smooth = 5 [0,1) [0,1)                          # Gaussian filter 5 bins
      smooth = 5 [1] [0,1) smooth = 5 [0,1) [1] } } }
```

to produce correlated beam description

3. Run WHIZARD with SINDARIN input:

```
beams = e1, E1 => circe2
$circe2_file = "ilc500.circe"
$circe2_design = "ILC"
?circe_polarized = false
```

Workflow GuineaPig/CIRCE2/WHIZARD

1. Run Guinea-Pig++ with

```
do_lumi=7;num_lumi=100000000;num_lumi_eg=100000000;num_lumi_gg=100000000;
```

   to produce lumi.[eg][eg].out with (E_1, E_2) pairs.

   [Large event numbers, as Guinea-Pig++ will produce only a small fraction!]

2. Run circe2_tool.opt with steering file

```
{ file="ilc500/beams.circe"                         # to be loaded by WHIZARD
  { design="ILC" roots=500 bins=100 scale=250 # E in [0,1]
    { pid/1=electron pid/2=positron pol=0         # unpolarized e-/e+
      events="ilc500/lumi.ee.out" columns=2    # <= Guinea-Pig
      lumi = 1564.763360                             # <= Guinea-Pig
      iterations = 10                                 # adapting bins
      smooth = 5 [0,1) [0,1)                         # Gaussian filter 5 bins
      smooth = 5 [1] [0,1) smooth = 5 [0,1) [1] } } }
```

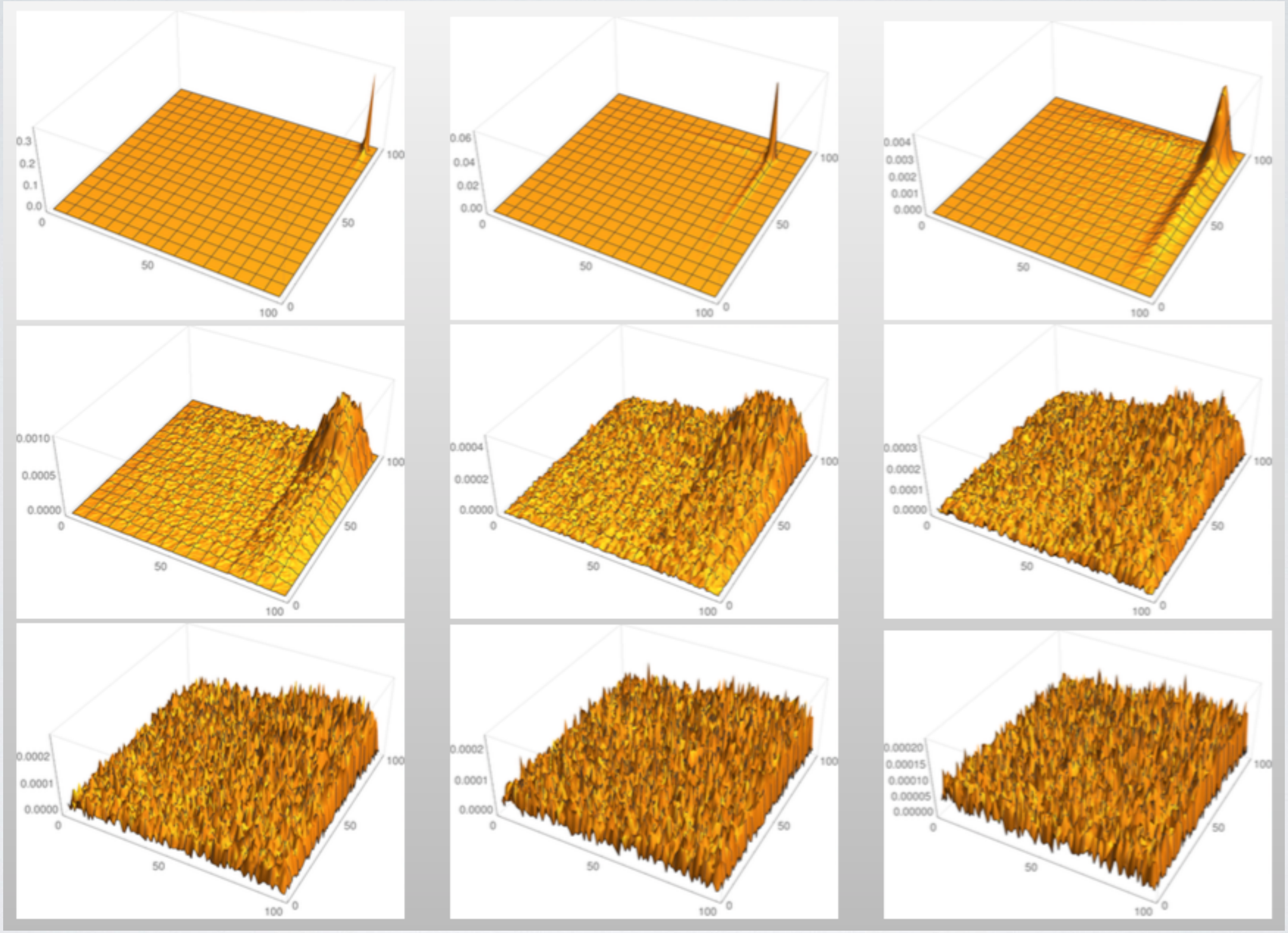
   to produce correlated beam description

3. Run WHIZARD with SINDARIN input:

```
beams = e1, E1 => circe2
$circe2_file = "ilc500.circe"
$circe2_design = "ILC"
?circe_polarized = false
```

polarized spectra on demand

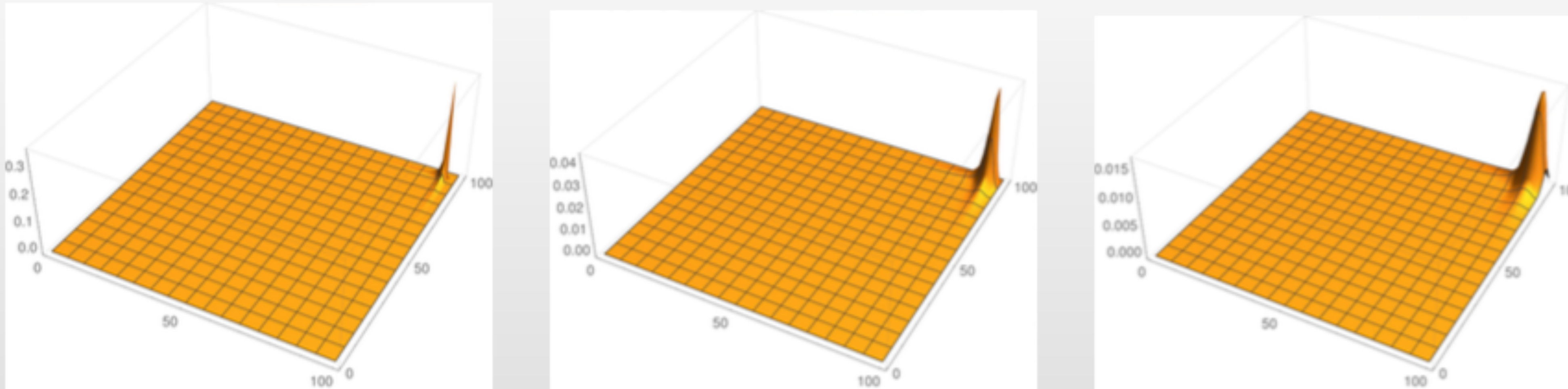
Iterations of Beam Spectrum



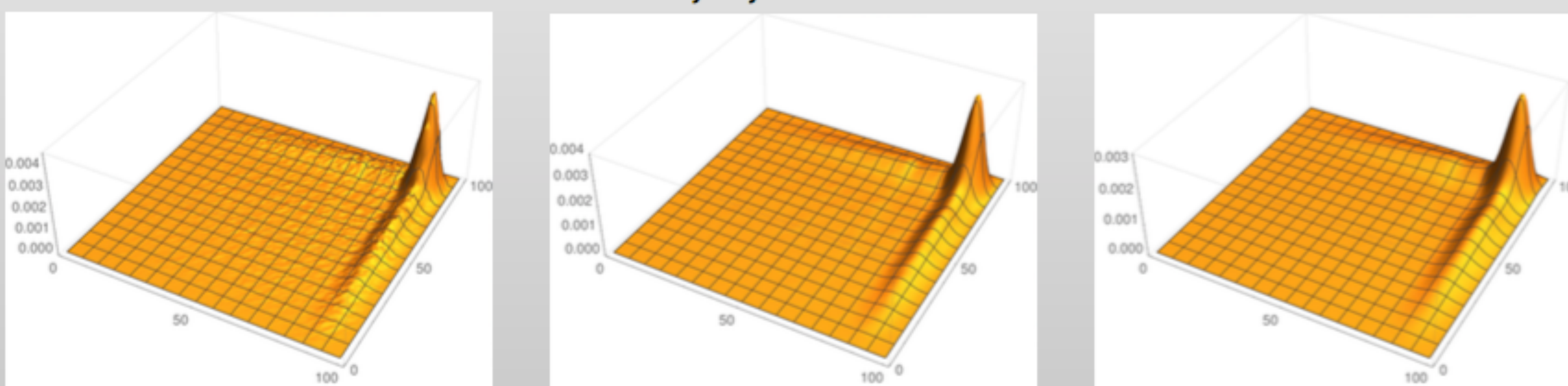
(171,306 GuineaPig events in 10,000 bins)

Iterations of Beam Spectrum

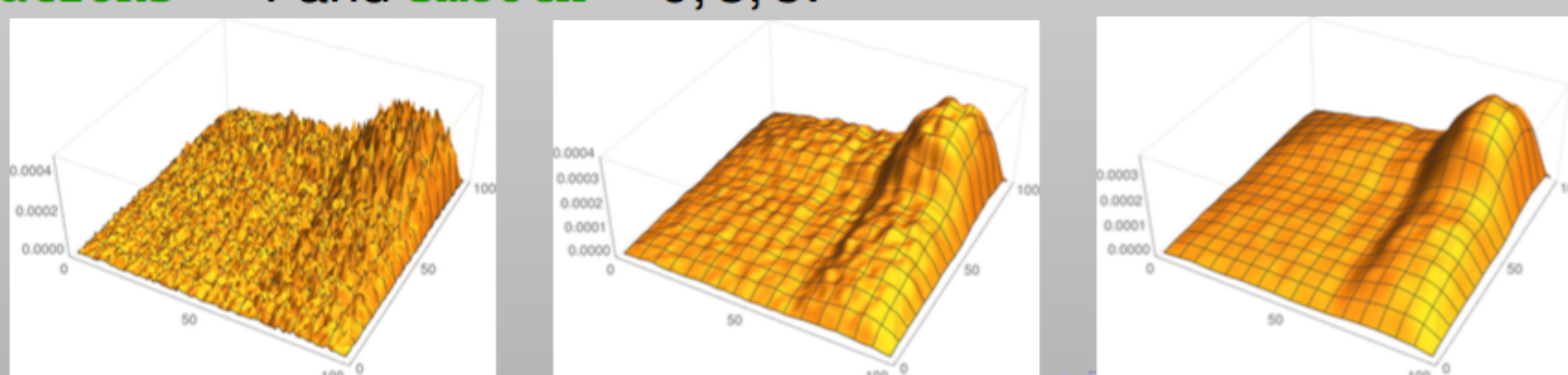
- $\texttt{iterations} = 0$ and $\texttt{smooth} = 0, 3, 5$:



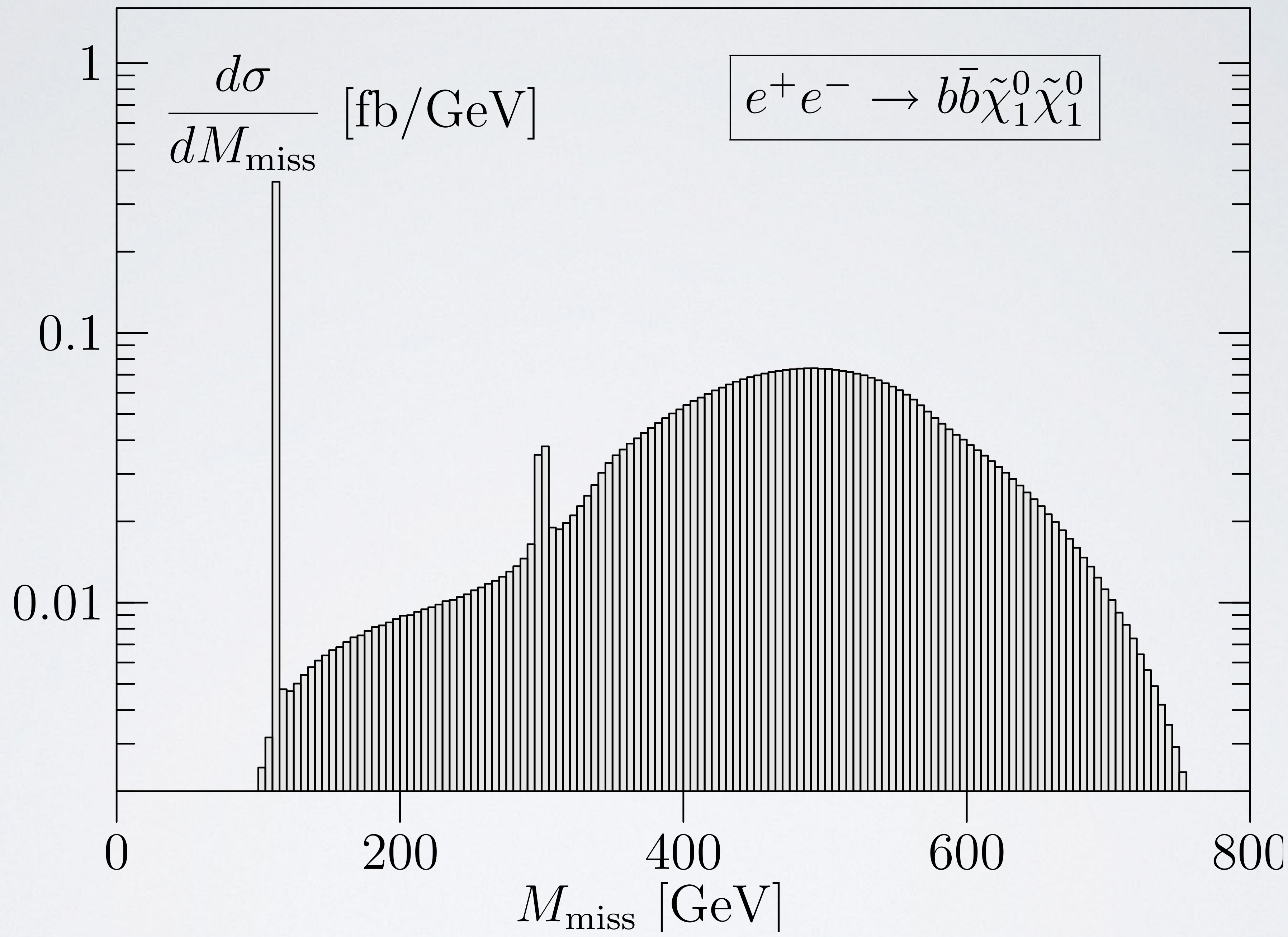
- $\texttt{iterations} = 2$ and $\texttt{smooth} = 0, 3, 5$:



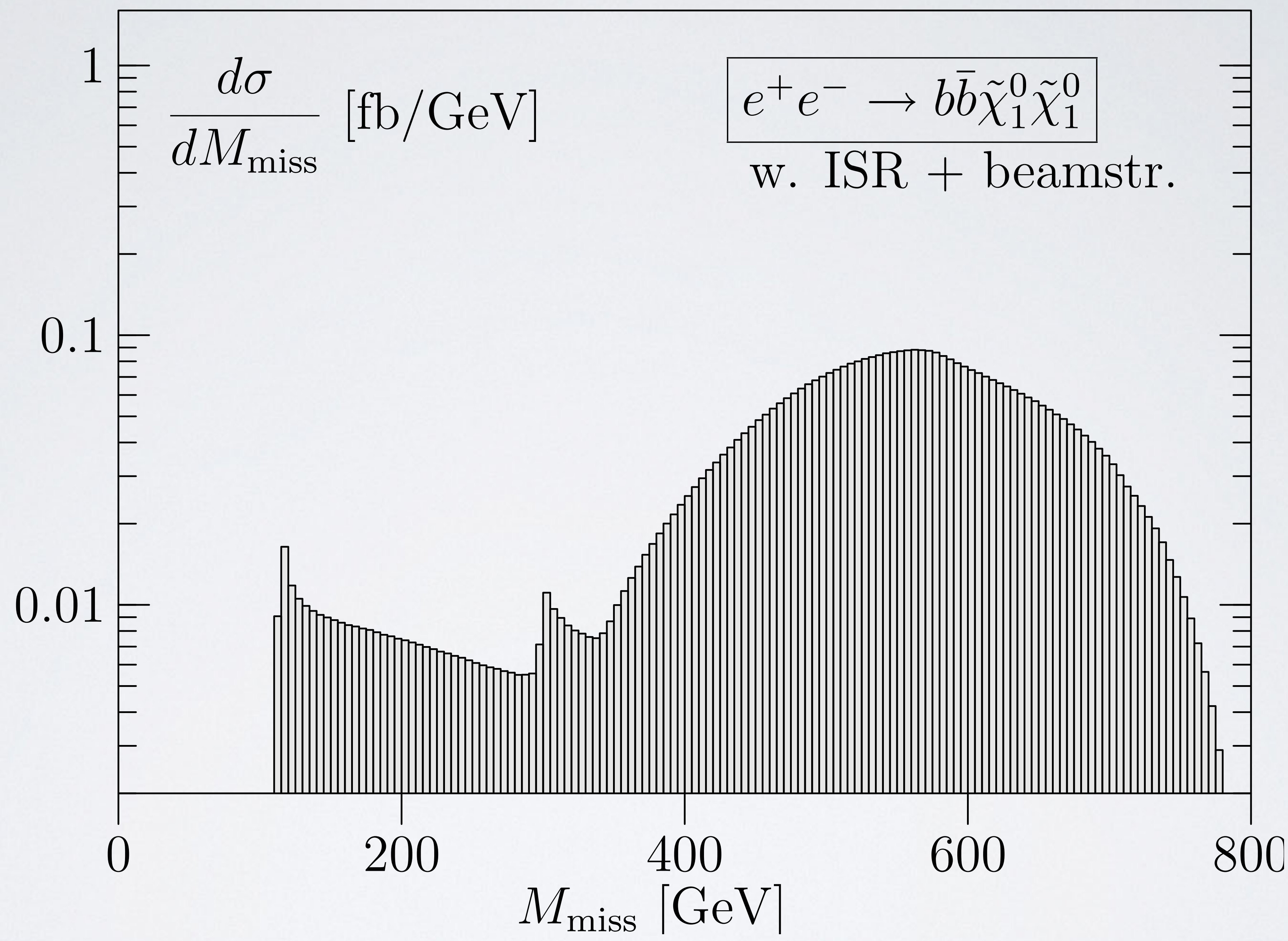
- $\texttt{iterations} = 4$ and $\texttt{smooth} = 0, 3, 5$:



Why care about beamstrahlung / ISR ?



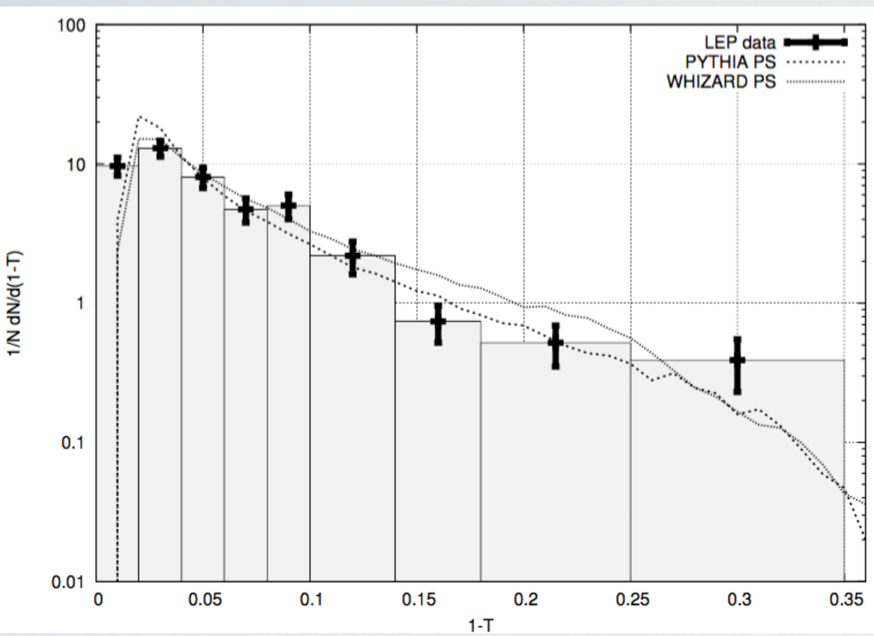
Why care about beamstrahlung / ISR ?

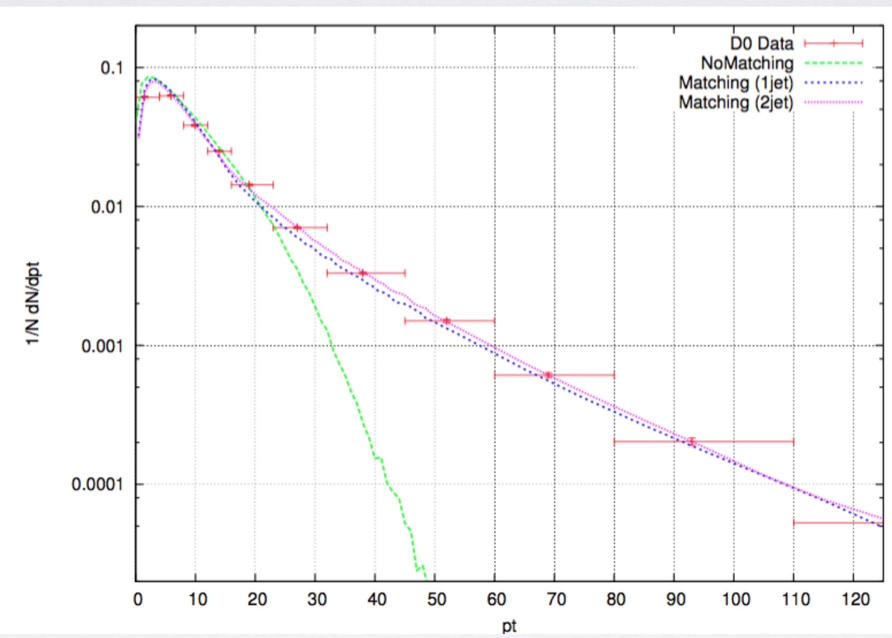


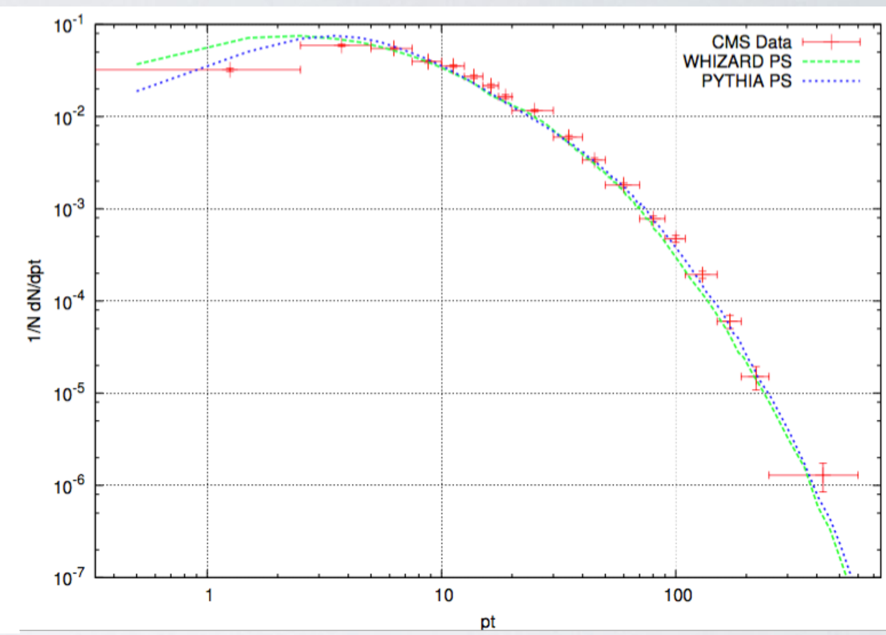
WHIZARD Parton Shower

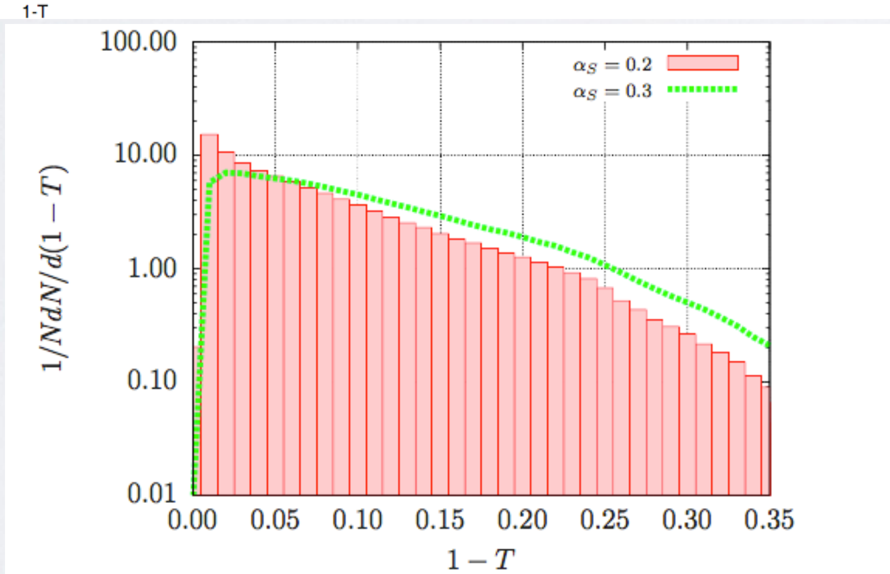
- Two independent implementations: kT-ordered QCD and Analytic QCD shower
- Analytic shower: no shower veto ⇒ exact shower history known, allows reweighting

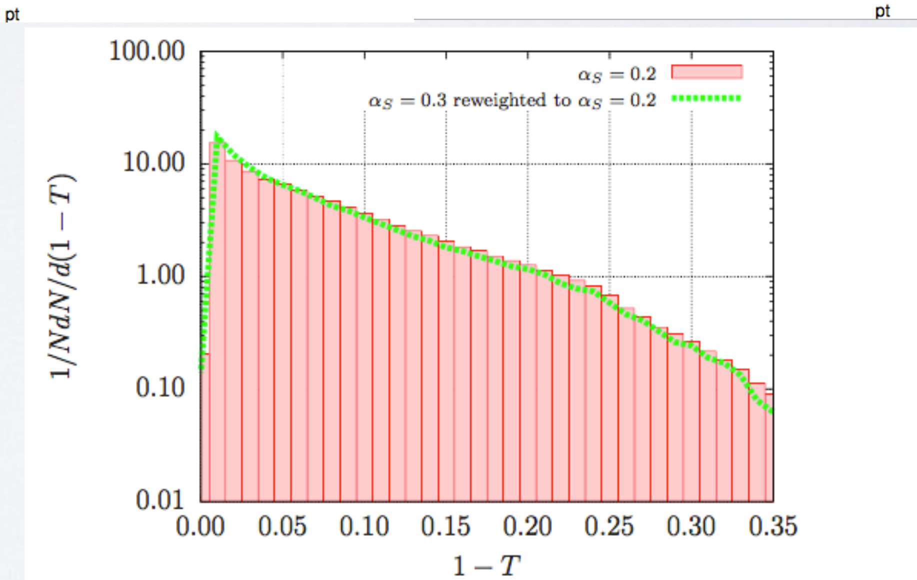
Kilian/JRR/Schmidt/Wiesler, JHEP **1204** 013 (2012)











- Technical overhaul of the shower / merging part
- Plans: implement GKS matching, QED shower (also interleaved, infrastructure ready)

Loops and Legs (and Cuts)

NLO Development in WHIZARD

- Need for precision predictions that match (sub-) percent experimental accuracy
- Scary challenge for the theory community [ok, we have some time still …]
- Mostly electroweak corrections, but also QCD and pure QED

Binoth Les Houches Interface (BLHA): Workflow

1. Process definition in SINDARIN (contract to One-Loop Program [OLP])
2. OLP generates code (Born/NLO interference), `WHIZARD` reads contract
3. NLO matrix element loaded into `WHIZARD`

NLO Development in WHIZARD

- Need for precision predictions that match (sub-) percent experimental accuracy
- Scary challenge for the theory community [ok, we have some time still …]
- Mostly electroweak corrections, but also QCD and pure QED

Binoth Les Houches Interface (BLHA): Workflow

1. Process definition in SINDARIN (contract to One-Loop Program [OLP])
2. OLP generates code (Born/NLO interference), `WHIZARD` reads contract
3. NLO matrix element loaded into `WHIZARD`

Working NLO interfaces to:

(first focus on QCD corrections)

- ⋆ GoSam [G. Cullen et al.]
- ⋆ OpenLoops [J. Lindert et al.]

NLO Development in WHIZARD

- Need for precision predictions that match (sub-) percent experimental accuracy
- Scary challenge for the theory community [ok, we have some time still …]
- Mostly electroweak corrections, but also QCD and pure QED

Binoth Les Houches Interface (BLHA): Workflow

1. Process definition in SINDARIN (contract to One-Loop Program [OLP])
2. OLP generates code (Born/NLO interference), WHIZARD reads contract
3. NLO matrix element loaded into WHIZARD

Working NLO interfaces to:

(first focus on QCD corrections)

- ★ GoSam [G. Cullen et al.]
- ★ OpenLoops [J. Lindert et al.]

WHIZARD v2.2.5 contains alpha version

QCD corrections (massless and massive emitters)

```
alpha_power = 2
alphas_power = 0

process eett = e1,E1 => t, tbar
     { nlo_calculation = “full” }
```

NLO Development in WHIZARD

- Need for precision predictions that match (sub-) percent experimental accuracy
- Scary challenge for the theory community [ok, we have some time still …]
- Mostly electroweak corrections, but also QCD and pure QED

Binoth Les Houches Interface (BLHA): Workflow

1. Process definition in SINDARIN (contract to One-Loop Program [OLP])
2. OLP generates code (Born/NLO interference), WHIZARD reads contract
3. NLO matrix element loaded into WHIZARD

Working NLO interfaces to:

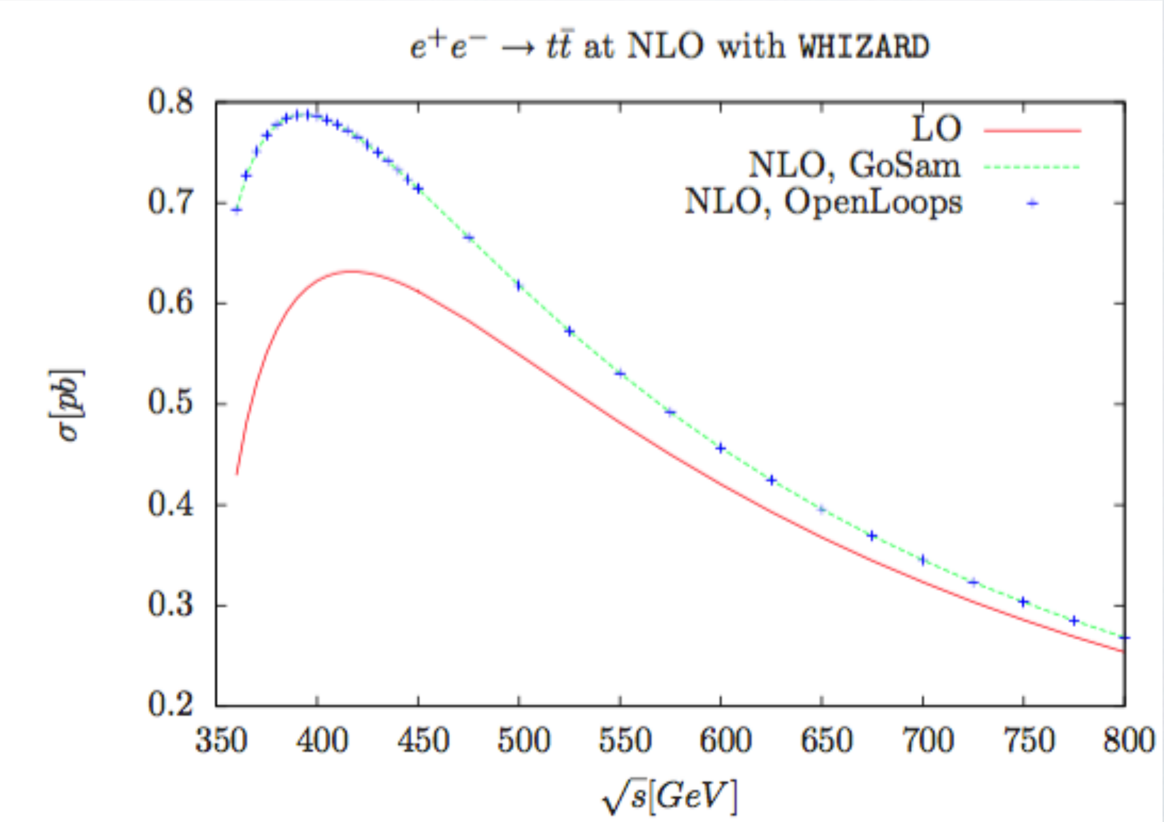
(first focus on QCD corrections)

- ★ GoSam [G. Cullen et al.]
- ★ OpenLoops [J. Lindert et al.]

WHIZARD v2.2.5 contains alpha version

QCD corrections (massless and massive emitters)

```
alpha_power = 2
alphas_power = 0

process eett = e1,E1 => t, tbar
     { nlo_calculation = “full” }
```



NLO Development in WHIZARD

- Need for precision predictions that match (sub-) percent experimental accuracy
- Scary challenge for the theory community [ok, we have some time still …]
- Mostly electroweak corrections, but also QCD and pure QED

Binoth Les Houches Interface (BLHA): Workflow

1. Process definition in SINDARIN (contract to One-Loop Program [OLP])
2. OLP generates code (Born/NLO interference), WHIZARD reads contract
3. NLO matrix element loaded into WHIZARD

Working NLO interfaces to:

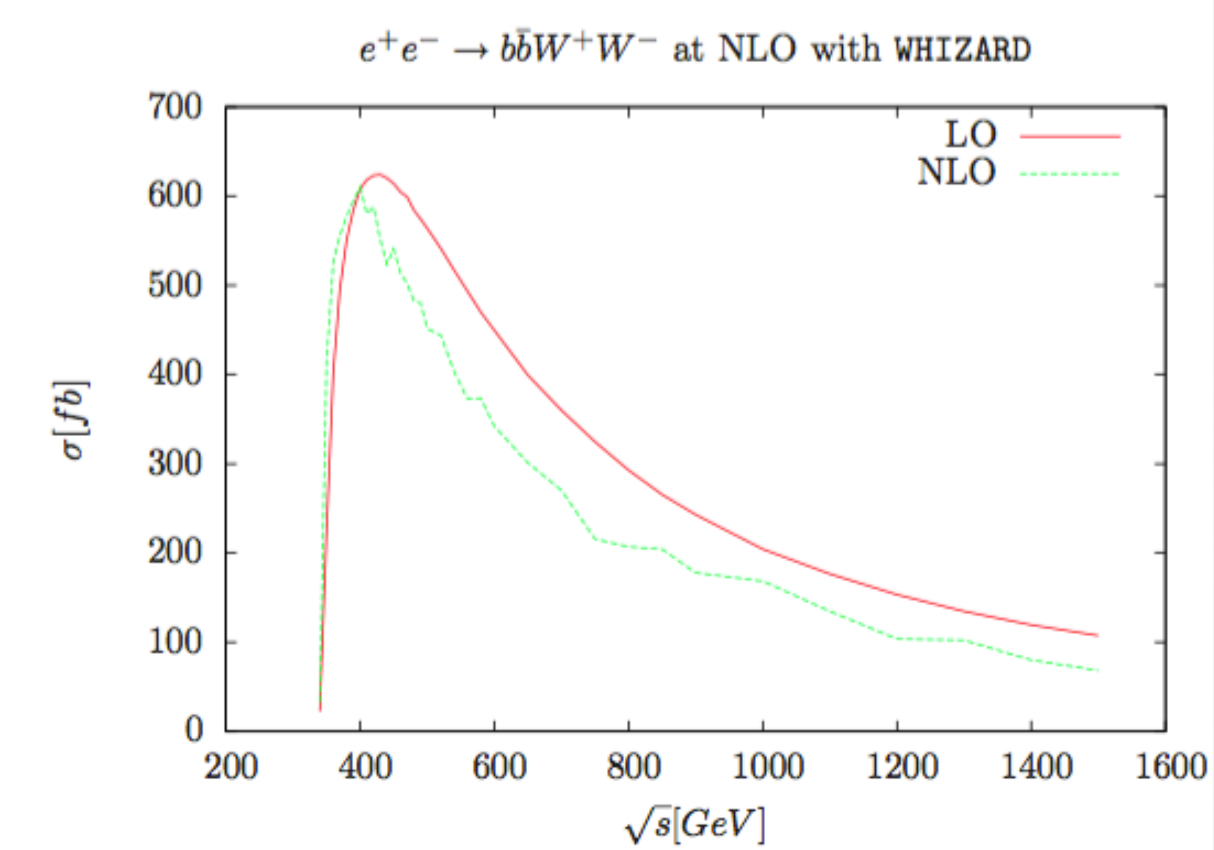
(first focus on QCD corrections)

- ★ GoSam [G. Cullen et al.]
- ★ OpenLoops [J. Lindert et al.]

WHIZARD v2.2.5 contains alpha version

QCD corrections (massless and massive emitters)

```
alpha_power = 2
alphas_power = 0

process eett = e1,E1 => t, tbar
     { nlo_calculation = “full” }
```



NLO Development in WHIZARD

- Need for precision predictions that match (sub-) percent experimental accuracy
- Scary challenge for the theory community [ok, we have some time still …]
- Mostly electroweak corrections, but also QCD and pure QED

Binoth Les Houches Interface (BLHA): Workflow

1. Process definition in SINDARIN (contract to One-Loop Program [OLP])
2. OLP generates code (Born/NLO interference), `WHIZARD` reads contract
3. NLO matrix element loaded into `WHIZARD`

Working NLO interfaces to:

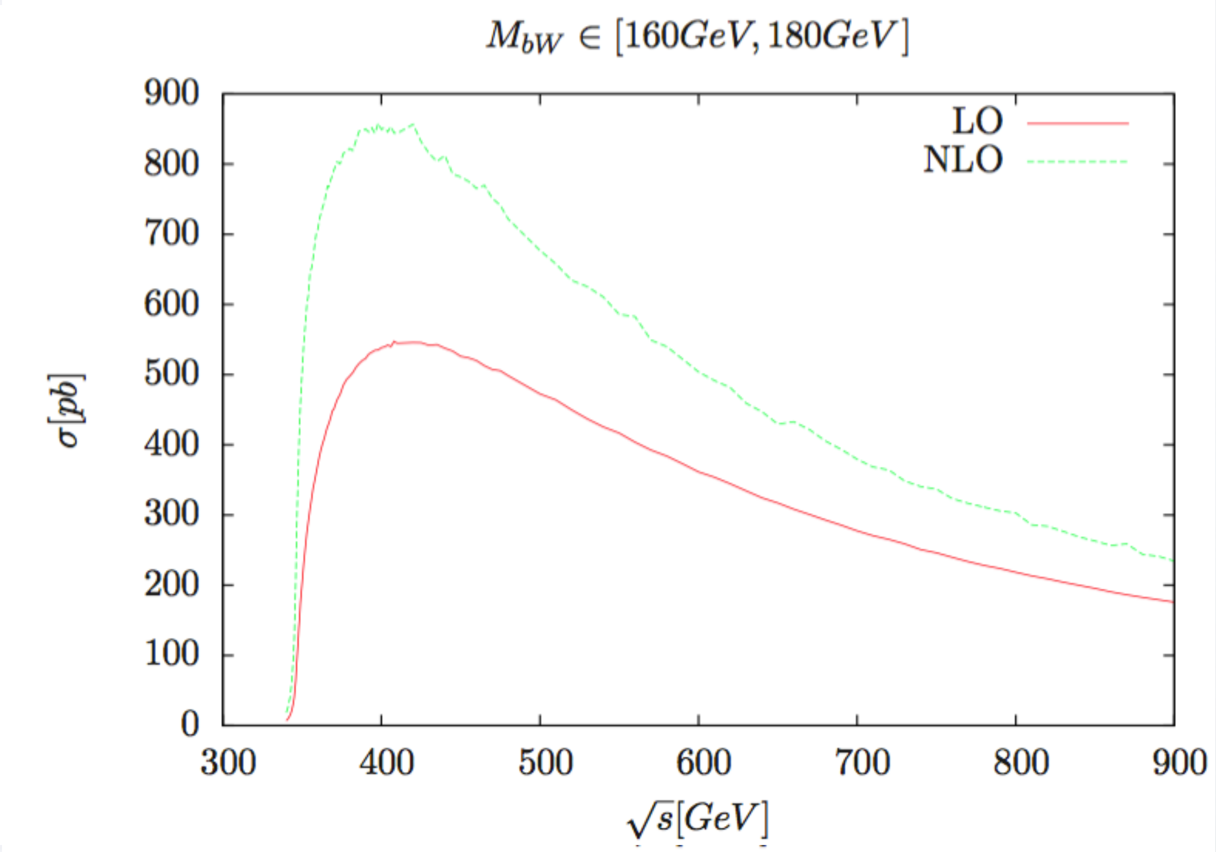
(first focus on QCD corrections)

- ★ GoSam [G. Cullen et al.]
- ★ OpenLoops [J. Lindert et al.]

WHIZARD v2.2.5 contains alpha version

QCD corrections (massless and massive emitters)

```
alpha_power = 2
alphas_power = 0

process eett = e1,E1 => t, tbar
     { nlo_calculation = "full" }
```



FKS Subtraction (Frixione/Kunszt/Signer)

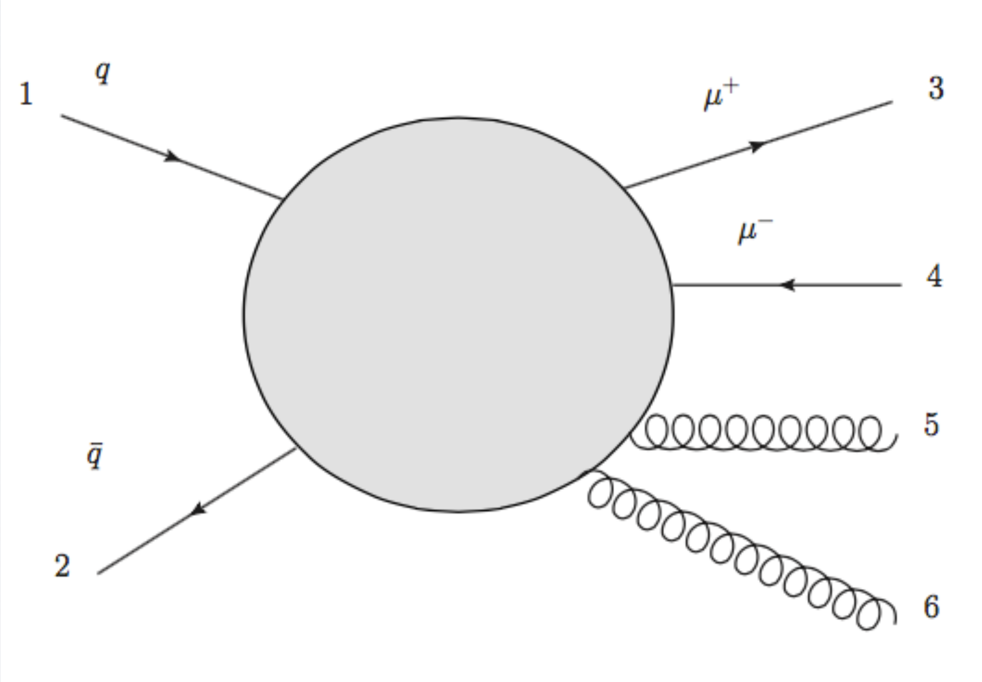
Subtraction formalism to make real and virtual contributions separately finite

$$d\sigma^{\text{NLO}} = \underbrace{\int_{n+1} \left(d\sigma^R - d\sigma^S\right)}_{\text{finite}} + \underbrace{\int_{n+1} d\sigma^S + \int_n d\sigma^V}_{\text{finite}}$$

FKS Subtraction (Frixione/Kunszt/Signer)

Subtraction formalism to make real and virtual contributions separately finite

$$d\sigma^{\mathrm{NLO}} = \underbrace{\int_{n+1} \left(d\sigma^R - d\sigma^S\right)}_{\text{finite}} + \underbrace{\int_{n+1} d\sigma^S + \int_n d\sigma^V}_{\text{finite}}$$



Automated Subtraction algorithm:

- Find all singular pairs

$$\mathcal{I} = \{(1,5),(1,6),(2,5),(2,6),(5,6)\}$$

- Partition phase space according to singular regions

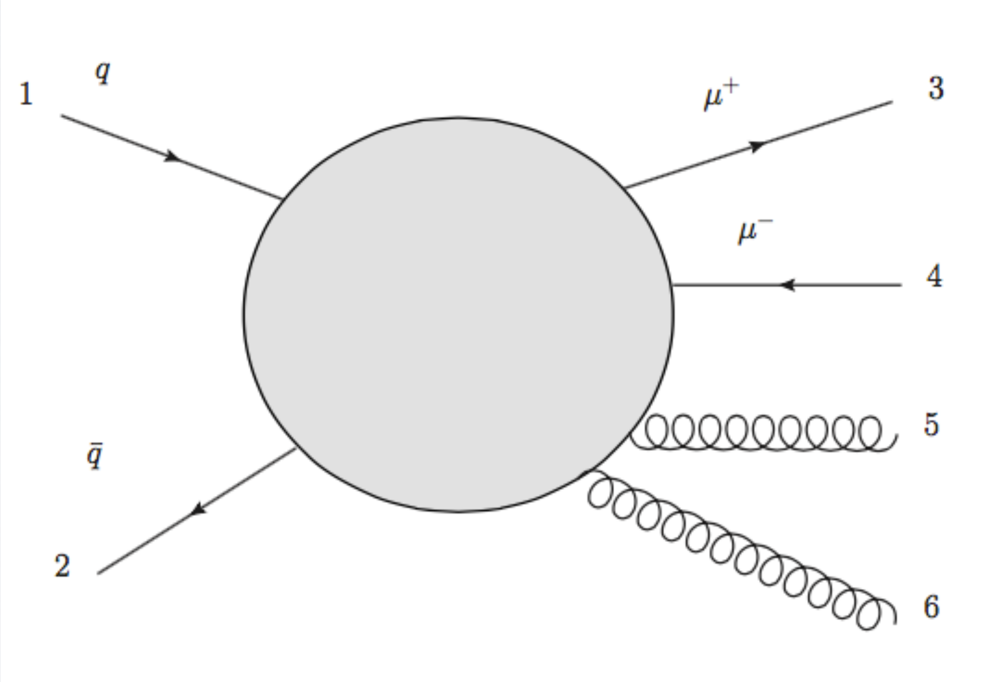
$$\mathbb{1} = \sum_{\alpha\in\mathcal{I}} S_\alpha(\Phi)$$

- Generate subtraction terms for singular regions

FKS Subtraction (Frixione/Kunszt/Signer)

Subtraction formalism to make real and virtual contributions separately finite

$$d\sigma^{\rm NLO} = \underbrace{\int_{n+1} \left(d\sigma^R - d\sigma^S\right)}_{\rm finite} + \underbrace{\int_{n+1} d\sigma^S + \int_n d\sigma^V}_{\rm finite}$$



Automated Subtraction algorithm:

- Find all singular pairs

$$\mathcal{I} = \{(1,5),(1,6),(2,5),(2,6),(5,6)\}$$

- Partition phase space according to singular regions

$$\mathbb{1} = \sum_{\alpha \in \mathcal{I}} S_\alpha(\Phi)$$

- Generate subtraction terms for singular regions

Soft subtraction involves color-correlated matrix elements:

$$\mathcal{B}_{kl} \sim -\sum_{\substack{\rm color\\ \rm spin}} \mathcal{A}^{(n)} \vec{\mathcal{Q}}(\mathcal{I}_k) \cdot \vec{\mathcal{Q}}(\mathcal{I}_l) \mathcal{A}^{(n)*},$$

Collinear subtraction involves spin-correlated matrix elements:

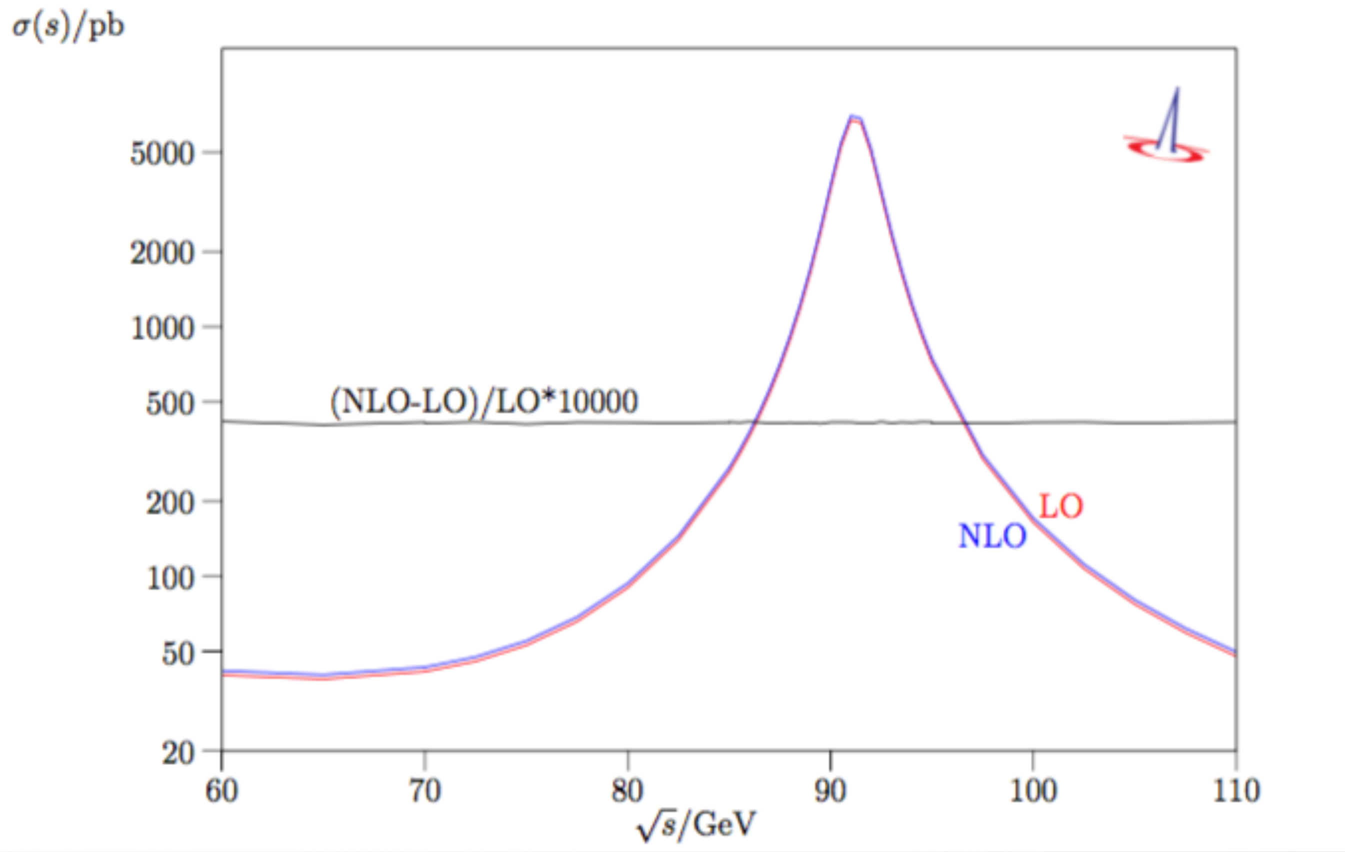
$$\mathcal{B}_{+-} \sim Re \left\{ \frac{\langle k_{\rm em} k_{\rm rad} \rangle}{[k_{\rm em} k_{\rm rad}]} \sum_{\substack{\rm color\\ \rm spin}} \mathcal{A}_+^{(n)} \mathcal{A}_-^{(n)*} \right\}$$

Examples and Validation

Simplest benchmark process:

$$e^+e^- \to q\bar{q} \quad \text{with} \quad \left(\sigma^{\text{NLO}} - \sigma^{\text{LO}}\right)/\sigma^{\text{LO}} = \alpha_s/\pi$$

Plot for total cross section for fixed strong coupling constant



List of validated QCD NLO processes

- $e^+e^- \to q\bar{q}$
- $e^+e^- \to q\bar{q}g$
- $e^+e^- \to \ell^+\ell^- q\bar{q}$
- $e^+e^- \to \ell^+\nu_\ell q\bar{q}$
- $e^+e^- \to t\bar{t}$
- $e^+e^- \to tW^-b$
- $e^+e^- \to W^+W^-b\bar{b}$
- $e^+e^- \to t\bar{t}H$

Caveat: no fixed-order NLO event generation due to missing counter-event infrastructure

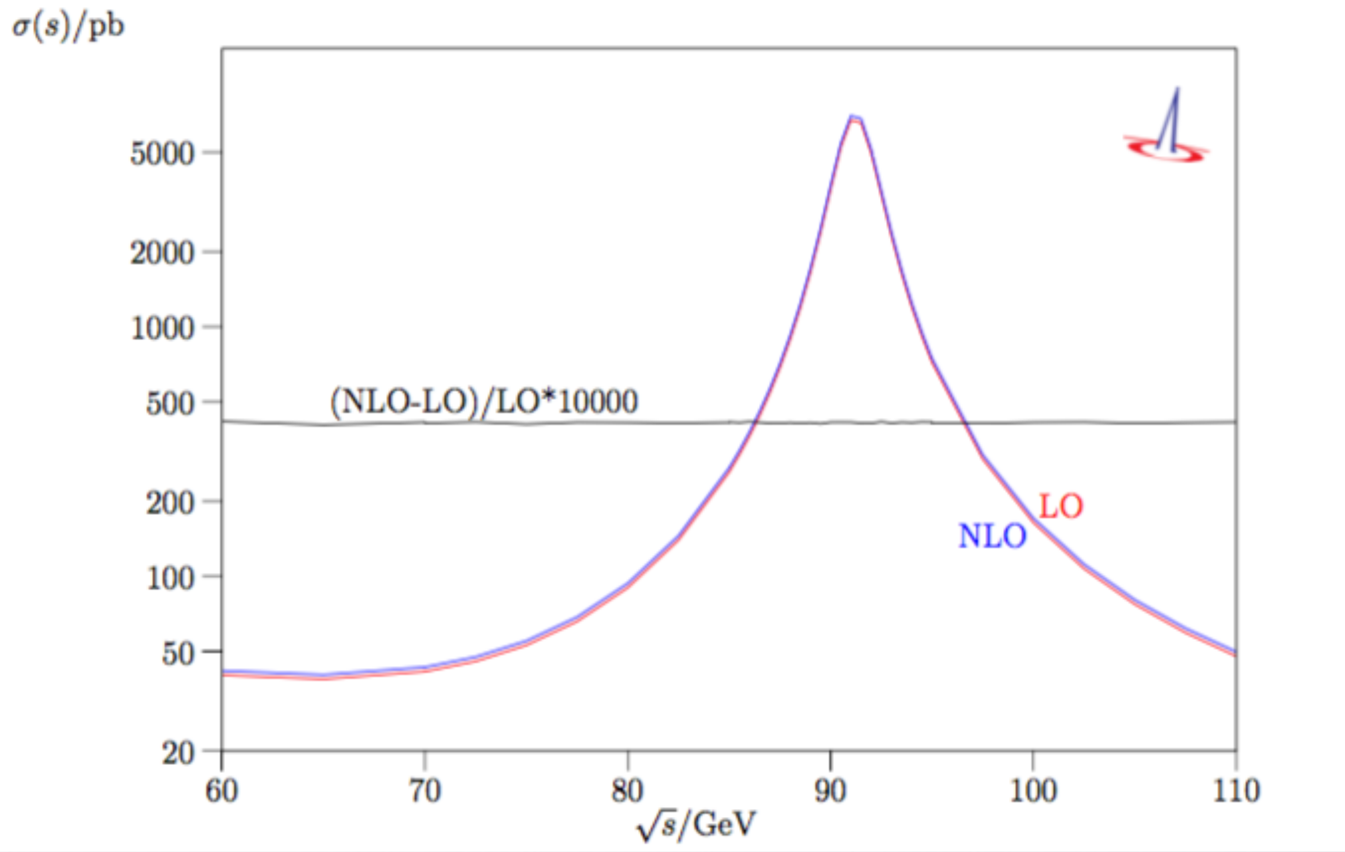
- Cross-checks with `MG5_aMC@NLO`

Examples and Validation

Simplest benchmark process:

$$e^+e^- \to q\bar{q} \quad \text{with} \quad \left(\sigma^{\rm NLO} - \sigma^{\rm LO}\right)/\sigma^{\rm LO} = \alpha_s/\pi$$

Plot for total cross section for fixed strong coupling constant



List of validated QCD NLO processes

- $e^+e^- \to q\bar{q}$
- $e^+e^- \to q\bar{q}g$
- $e^+e^- \to \ell^+\ell^- q\bar{q}$
- $e^+e^- \to \ell^+\nu_\ell q\bar{q}$
- $e^+e^- \to t\bar{t}$
- $e^+e^- \to tW^-b$
- $e^+e^- \to W^+W^-b\bar{b}$
- $e^+e^- \to t\bar{t}H$

Caveat: no fixed-order NLO event generation due to missing counter-event infrastructure

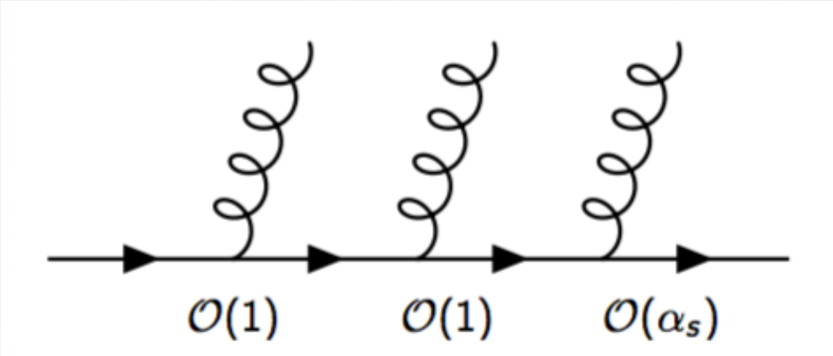
- Cross-checks with `MG5_aMC@NLO`

✦ QCD NLO infrastructure in pp finished yesterday

✦ First attempts on electroweak corrections, interfacing the `RECOLA` code [Denner et al.]

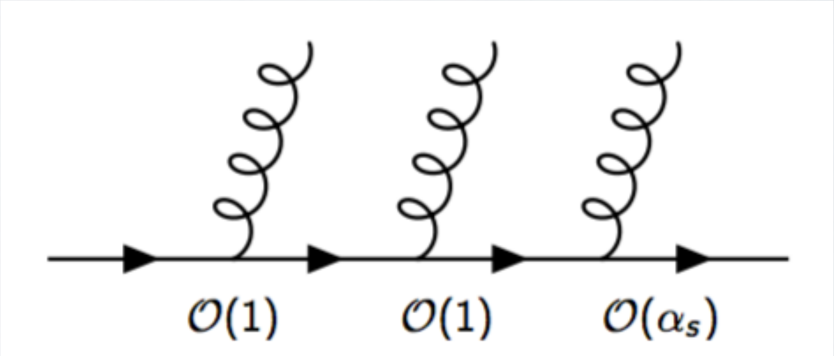
POWHEG Matching in WHIZARD

- Soft gluon emission before hard emission generate large logs
- Perturbative α_s : $|\mathcal{M}_{\rm soft}|^2 \sim \frac{1}{k_T^2} \to \log \frac{k_T^{\rm max}}{k_T^{\rm min}}$
- Matrix element + parton shower has to take this into account
- POWHEG method: hardest emission first [Nason et al.]



POWHEG Matching in WHIZARD

- Soft gluon emission before hard emission generate large logs
- Perturbative αs : $|\mathcal{M}_{\rm soft}|^2 \sim \frac{1}{k_T^2} \to \log \frac{k_T^{\max}}{k_T^{\min}}$
- Matrix element + parton shower has to take this into account
- POWHEG method: hardest emission first [Nason et al.]



- Complete NLO events

$$\overline{B}(\Phi_n) = B(\Phi_n) + V(\Phi_n) + \int d\Phi_{\rm rad} R(\Phi_{n+1})$$

- POWHEG generate events according to the formula:

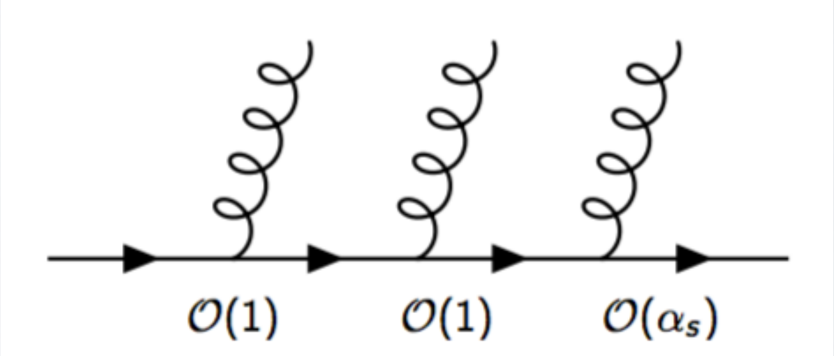
$$d\sigma = \overline{B}(\Phi_n) \left[\Delta_R^{\rm NLO}(k_T^{\min}) + \Delta_R^{\rm NLO}(k_T) \frac{R(\Phi_{n+1})}{B(\Phi_n)} d\Phi_{\rm rad} \right]$$

- Uses the modified Sudakov form factor:

$$\Delta_R^{\rm NLO}(k_T) = \exp\left[- \int d\Phi_{\rm rad} \frac{R(\Phi_{n+1})}{B(\Phi_n)} \theta(k_T(\Phi_{n+1}) - k_T) \right]$$

POWHEG Matching in WHIZARD

- Soft gluon emission before hard emission generate large logs
- Perturbative α_s : $|\mathcal{M}_{\rm soft}|^2 \sim \frac{1}{k_T^2} \to \log \frac{k_T^{\rm max}}{k_T^{\rm min}}$
- Matrix element + parton shower has to take this into account
- POWHEG method: hardest emission first [Nason et al.]



- Complete NLO events

$$\overline{B}(\Phi_n) = B(\Phi_n) + V(\Phi_n) + \int d\Phi_{\rm rad} R(\Phi_{n+1})$$

- POWHEG generate events according to the formula:

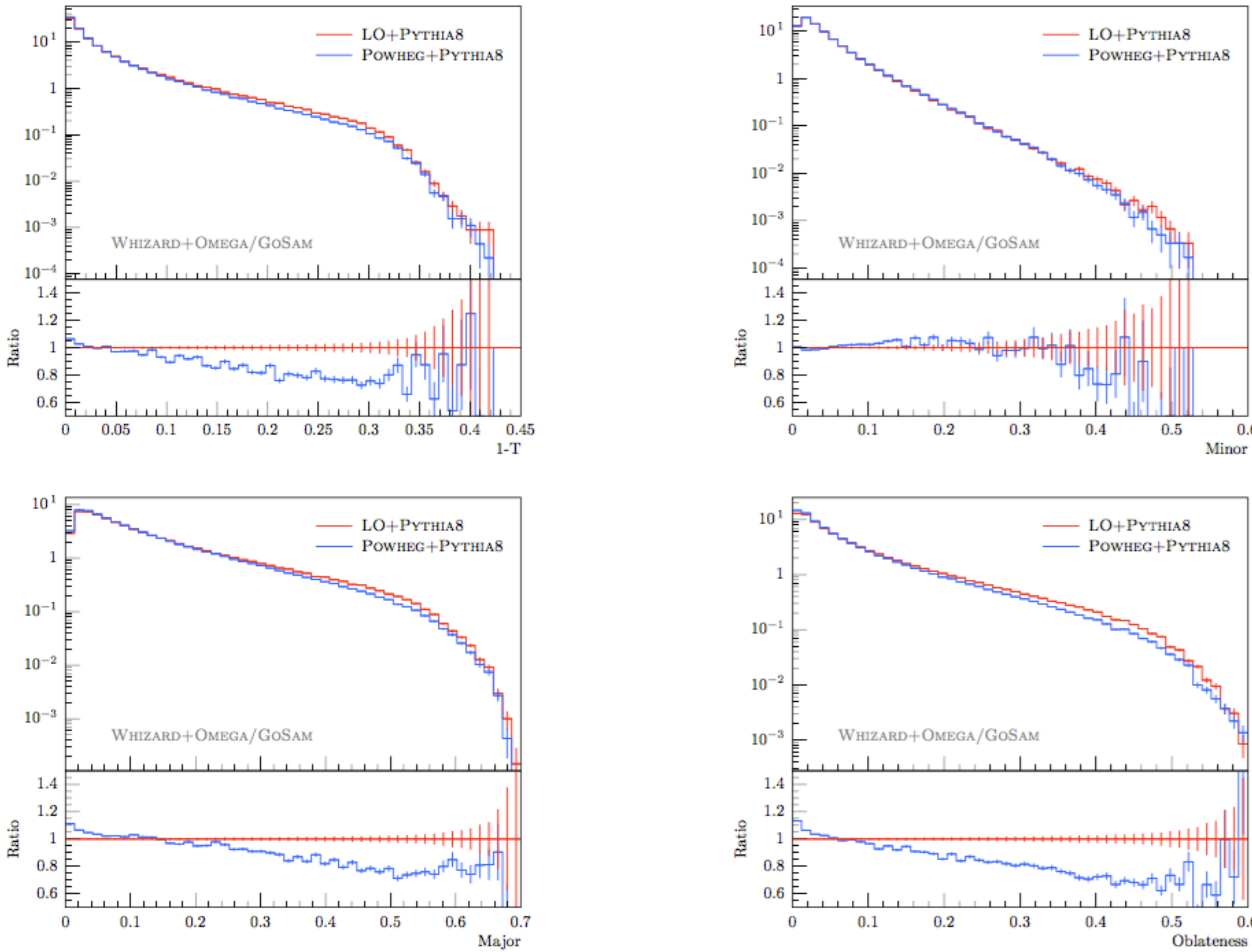
$$d\sigma = \overline{B}(\Phi_n) \left[\Delta_R^{\rm NLO}(k_T^{\rm min}) + \Delta_R^{\rm NLO}(k_T) \frac{R(\Phi_{n+1})}{B(\Phi_n)} d\Phi_{\rm rad} \right]$$

- Uses the modified Sudakov form factor:

$$\Delta_R^{\rm NLO}(k_T) = \exp\left[- \int d\Phi_{\rm rad} \frac{R(\Phi_{n+1})}{B(\Phi_n)} \theta(k_T(\Phi_{n+1}) - k_T) \right]$$

- Hardest emission: $k_T^{\rm max}$; shower with imposing a veto:
- $\overline{B} < 0$ if virtual and real terms larger than Born: shouldn't happen in perturbative regions
- Reweighting such that $\overline{B} > 0$ for all events
- POWHEG: Positive Weight Hardest Emission Generator now implemented in `WHIZARD`

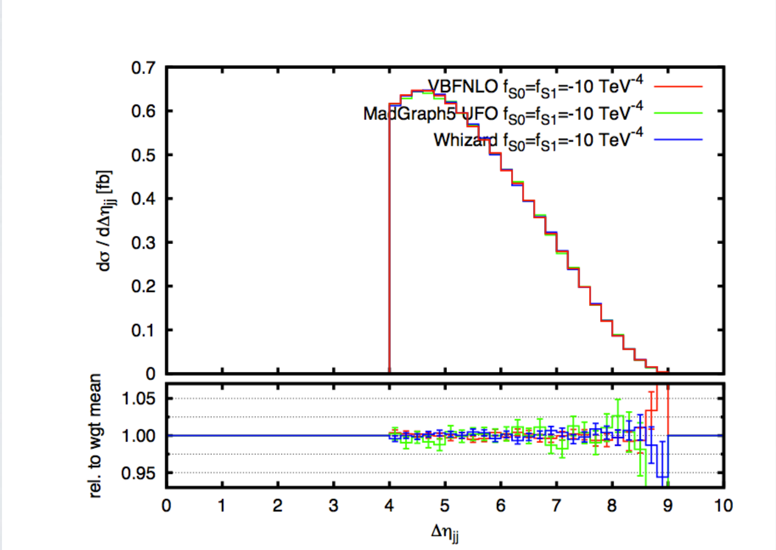
POWHEG Matching in e+e- to dijets

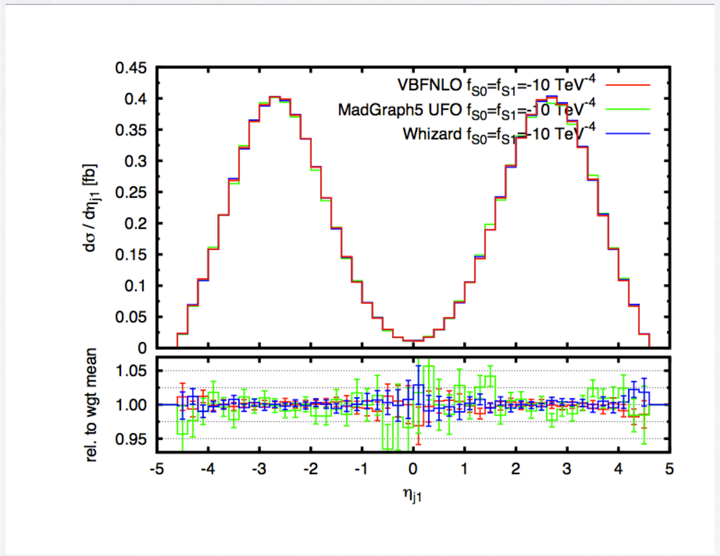


New Physics in Vector Boson Scattering

- Vector Boson Scattering (VBS) major measurement of LHC runs II/III Gianotti, CERN 01/2014
- Light Higgs suppression makes VBS prime candidate for BSM searches
- Model-independent EFT descriptions (almost) useless: either weakly-coupled resonances in reach or strongly-coupled sectors Alboteanu/Kilian/JRR, 2008; Kilian/Ohl/JRR/Sekulla, 2014
- Parameterize new physics by dim 6/dim 8 operators, calculate unitarity limits
- K-matrix unitarization implemented in `WHIZARD` (both for operators and resonances)

For the pure operators: full agreement between `WHIZARD, Madgraph5, VBFNLO`





New Physics in Vector Boson Scattering

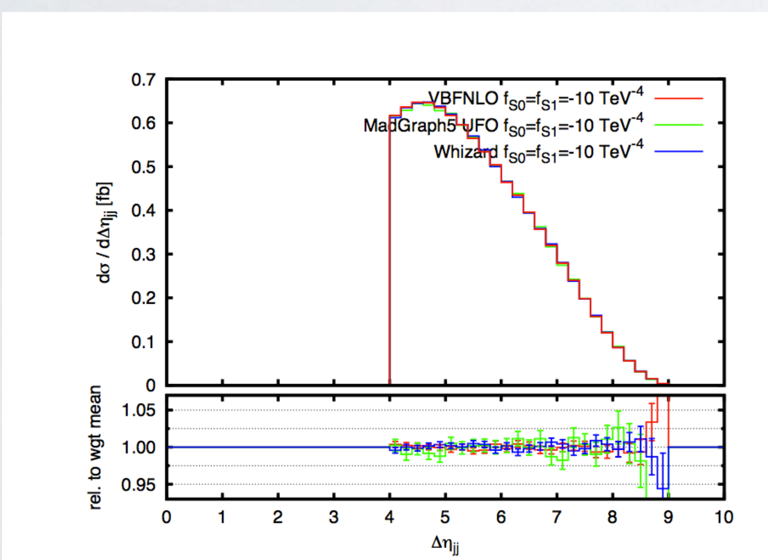
- Vector Boson Scattering (VBS) major measurement of LHC runs II/III Gianotti, CERN 01/2014
- Light Higgs suppression makes VBS prime candidate for BSM searches
- Model-independent EFT descriptions (almost) useless: either weakly-coupled resonances in reach or strongly-coupled sectors Alboteanu/Kilian/JRR, 2008; Kilian/Ohl/JRR/Sekulla, 2014
- Parameterize new physics by dim 6/dim 8 operators, calculate unitarity limits
- K-matrix unitarization implemented in `WHIZARD` (both for operators and resonances)

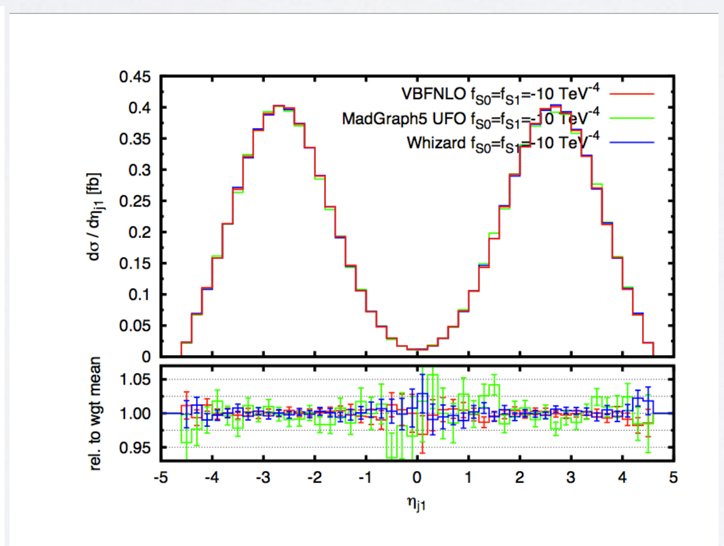
$$\mathcal{L}_{HD} = F_{HD} \ \mathrm{tr}\left[\mathbf{H}^\dagger\mathbf{H} - \frac{v^2}{4}\right] \cdot \mathrm{tr}\left[(\mathbf{D}_\mu\mathbf{H})^\dagger(\mathbf{D}^\mu\mathbf{H})\right]$$

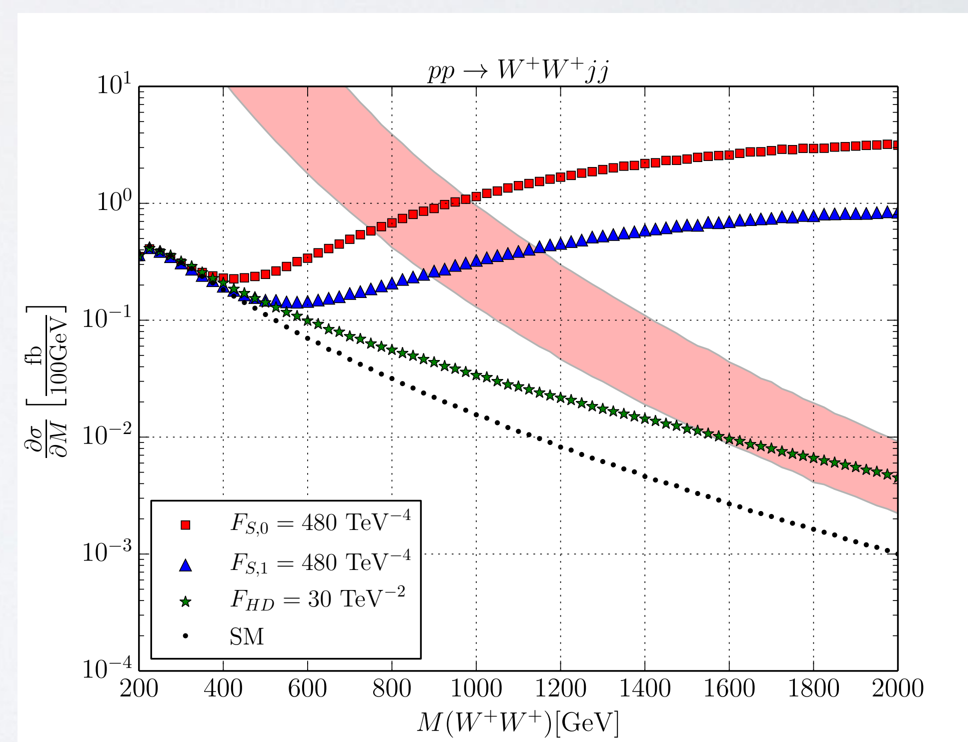
$$\mathcal{L}_{S,0} = F_{S,0} \ \mathrm{tr}\left[(\mathbf{D}_\mu\mathbf{H})^\dagger\mathbf{D}_\nu\mathbf{H}\right] \cdot \mathrm{tr}\left[(\mathbf{D}^\mu\mathbf{H})^\dagger\mathbf{D}^\nu\mathbf{H}\right]$$

$$\mathcal{L}_{S,1} = F_{S,1} \ \mathrm{tr}\left[(\mathbf{D}_\mu\mathbf{H})^\dagger\mathbf{D}^\mu\mathbf{H}\right] \cdot \mathrm{tr}\left[(\mathbf{D}_\nu\mathbf{H})^\dagger\mathbf{D}^\nu\mathbf{H}\right]$$

For the pure operators: full agreement between `WHIZARD, Madgraph5, VBFNLO`







New Physics in Vector Boson Scattering

- UV-incomplete amplitudes could violate perturbative (tree-level) unitarity
- Algorithm: diagonalize the *S*-matrix by using spin-isospin eigenamplitudes (for on-shell (electroweak) vector bosons $\mathcal{A}(s,t,u) = 32\pi \sum_\ell (2\ell+1)\mathcal{A}_\ell(s) P_\ell(1+2t/s)$

$$\mathcal{A}(w^+w^- \to zz) = A(s,t,u),$$
$$\mathcal{A}(zz \to zz) = A(s,t,u) + A(t,s,u) + A(u,s,t),$$
$$\mathcal{A}(w^+w^- \to w^+w^-) = A(s,t,u) + A(t,s,u),$$
$$\mathcal{A}(w^+z \to w^+z) = A(t,s,u),$$
$$\mathcal{A}(w^+w^+ \to w^+w^+) = A(t,s,u) + A(u,s,t),$$

- Unitarization for longitudinal modes (Goldstone bosons)
- Transversal modes could also violate unitarity (ignored for now)
- Project back on (off-shell) bosons

New Physics in Vector Boson Scattering

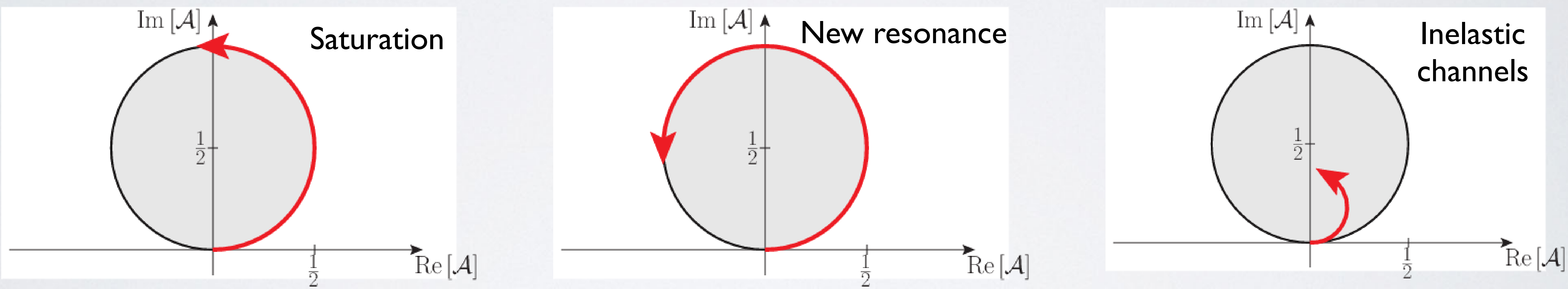
- UV-incomplete amplitudes could violate perturbative (tree-level) unitarity
- Algorithm: diagonalize the S-matrix by using spin-isospin eigenamplitudes (for on-shell (electroweak) vector bosons $\mathcal{A}(s,t,u) = 32\pi \sum_\ell (2\ell+1)\mathcal{A}_\ell(s) P_\ell(1+2t/s)$

$$\mathcal{A}(w^+w^- \to zz) = A(s,t,u),$$
$$\mathcal{A}(zz \to zz) = A(s,t,u) + A(t,s,u) + A(u,s,t),$$
$$\mathcal{A}(w^+w^- \to w^+w^-) = A(s,t,u) + A(t,s,u),$$
$$\mathcal{A}(w^+z \to w^+z) = A(t,s,u),$$
$$\mathcal{A}(w^+w^+ \to w^+w^+) = A(t,s,u) + A(u,s,t),$$

- Unitarization for longitudinal modes (Goldstone bosons)
- Transversal modes could also violate unitarity (ignored for now)
- Project back on (off-shell) bosons

Alboteanu/Kilian/JRR, 2008; Kilian/Ohl/JRR/Sekulla, 2014

Possible behavior:



New Physics in Vector Boson Scattering

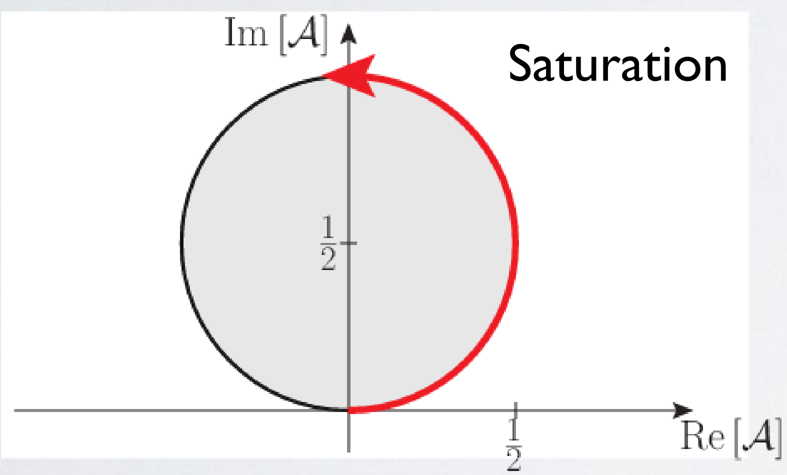
- UV-incomplete amplitudes could violate perturbative (tree-level) unitarity
- Algorithm: diagonalize the S-matrix by using spin-isospin eigenamplitudes (for on-shell (electroweak) vector bosons $\mathcal{A}(s,t,u) = 32\pi \sum_\ell (2\ell+1)\mathcal{A}_\ell(s) P_\ell(1+2t/s)$

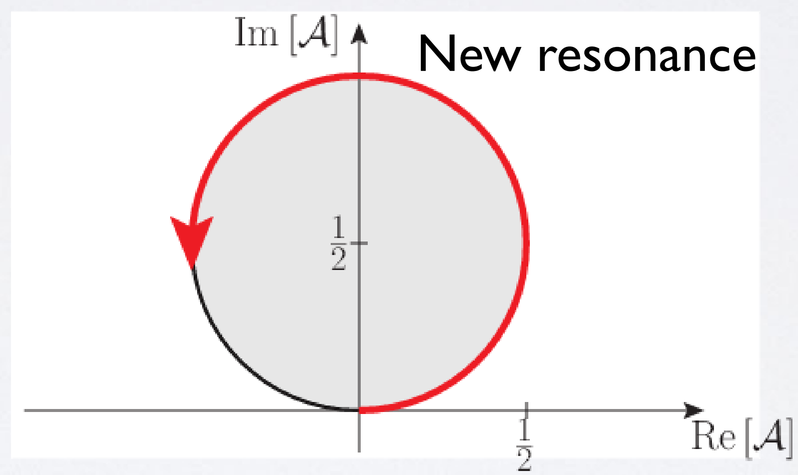
$$\mathcal{A}(w^+w^- \to zz) = A(s,t,u),$$
$$\mathcal{A}(zz \to zz) = A(s,t,u) + A(t,s,u) + A(u,s,t),$$
$$\mathcal{A}(w^+w^- \to w^+w^-) = A(s,t,u) + A(t,s,u),$$
$$\mathcal{A}(w^+z \to w^+z) = A(t,s,u),$$
$$\mathcal{A}(w^+w^+ \to w^+w^+) = A(t,s,u) + A(u,s,t),$$

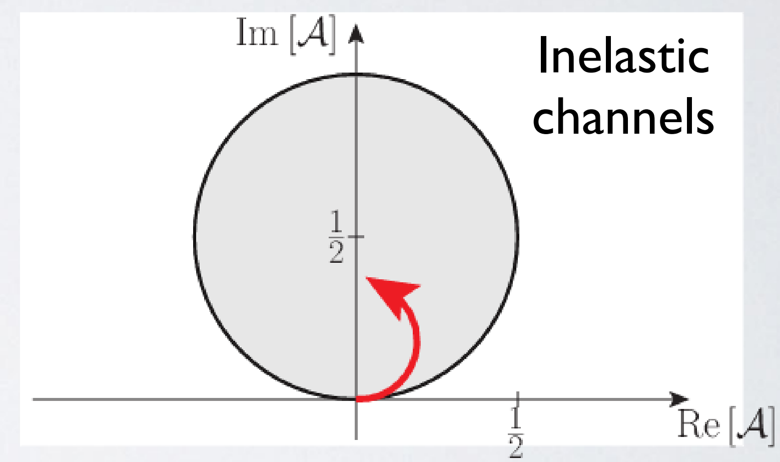
- Unitarization for longitudinal modes (Goldstone bosons)
- Transversal modes could also violate unitarity (ignored for now)
- Project back on (off-shell) bosons

Alboteanu/Kilian/JRR, 2008; Kilian/Ohl/JRR/Sekulla, 2014

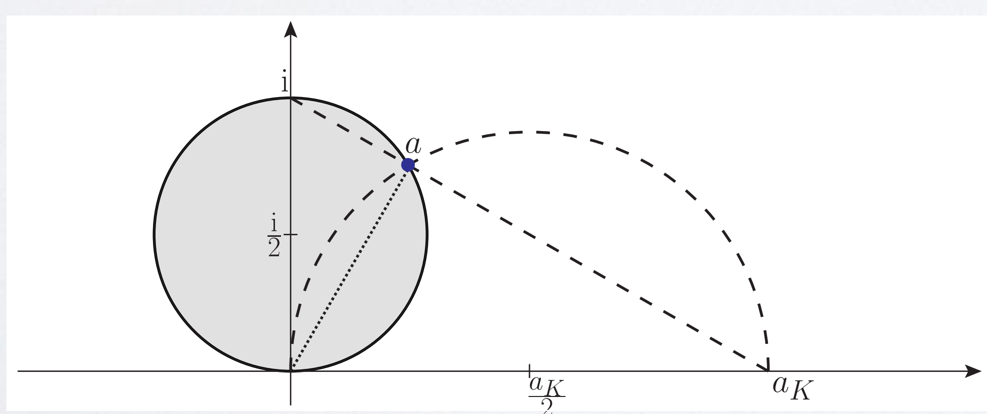
Possible behavior:

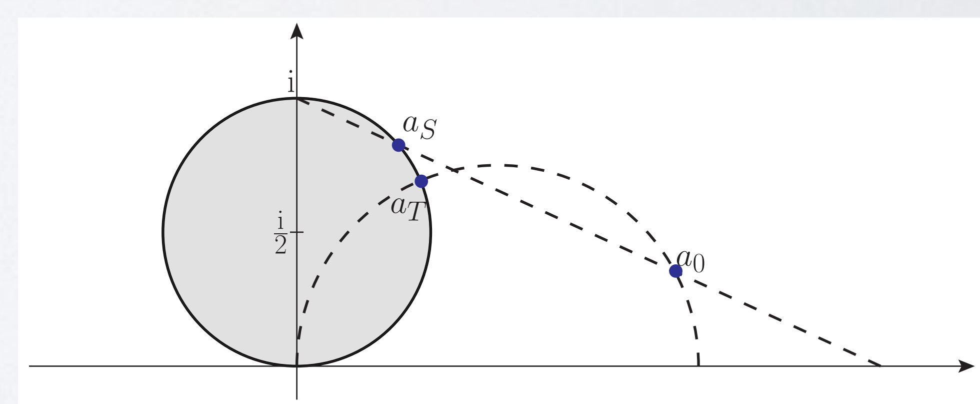




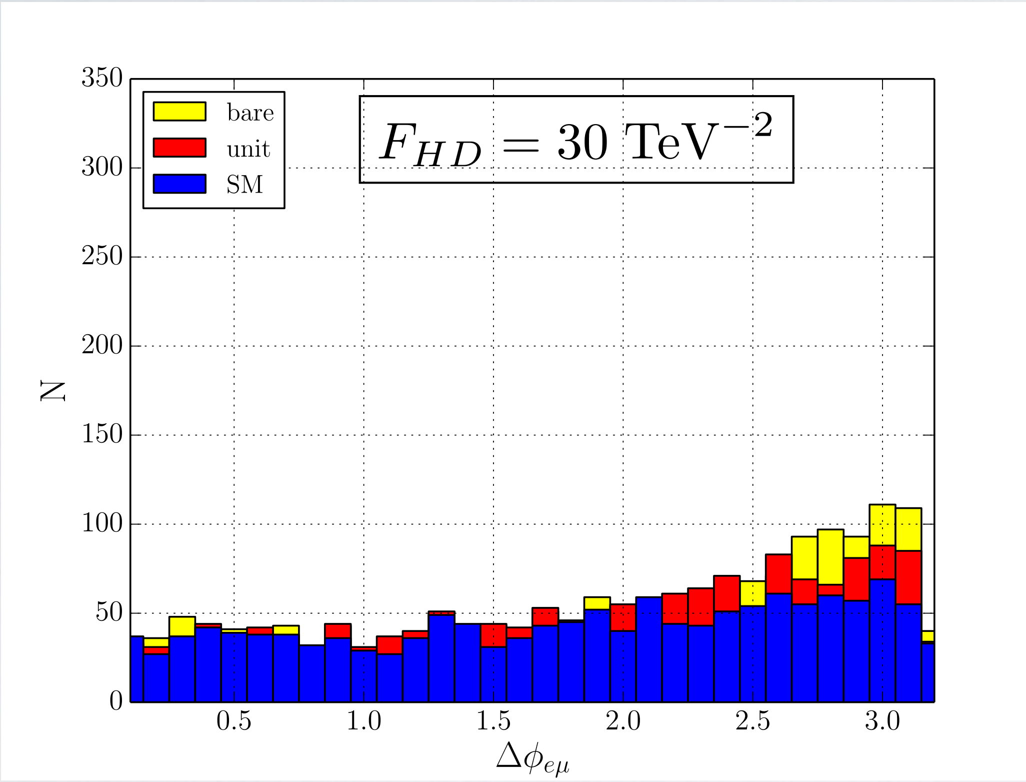


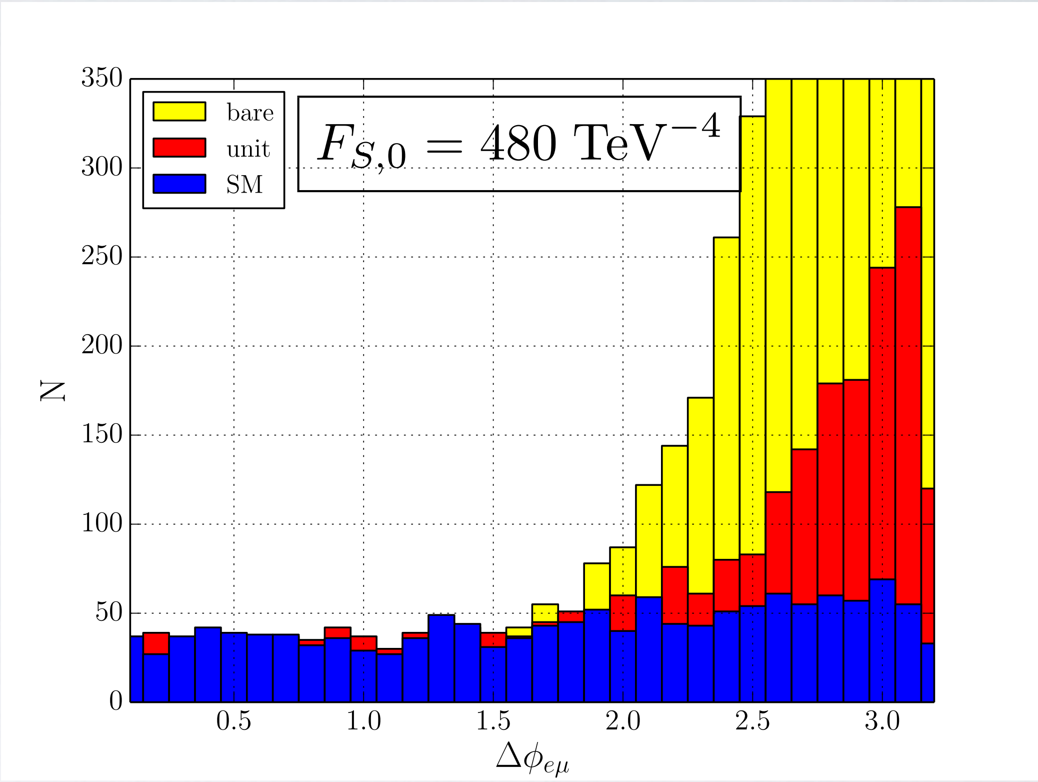
K-matrix vs. T-matrix unitarization

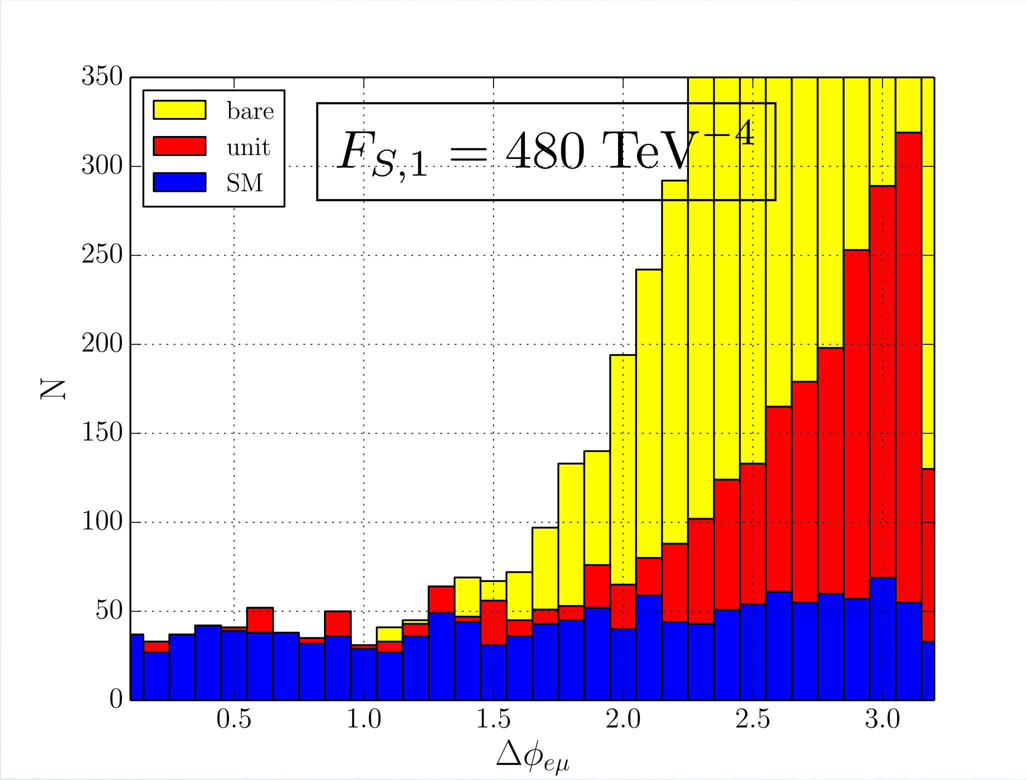




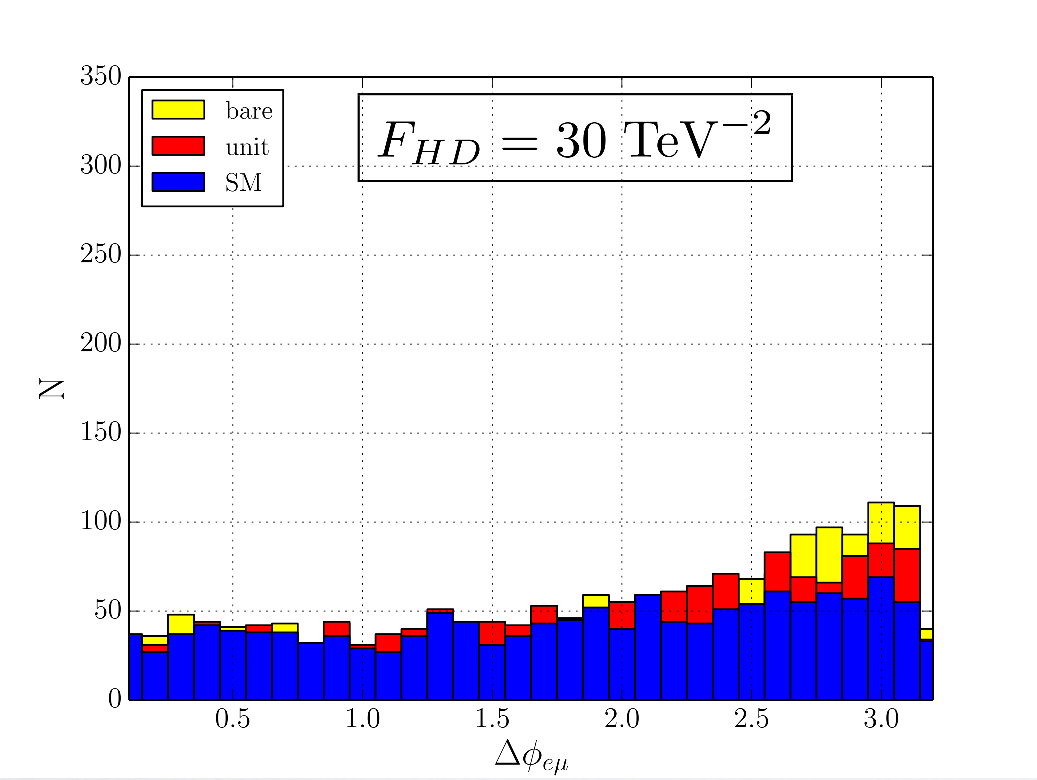
New Physics in Vector Boson Scattering

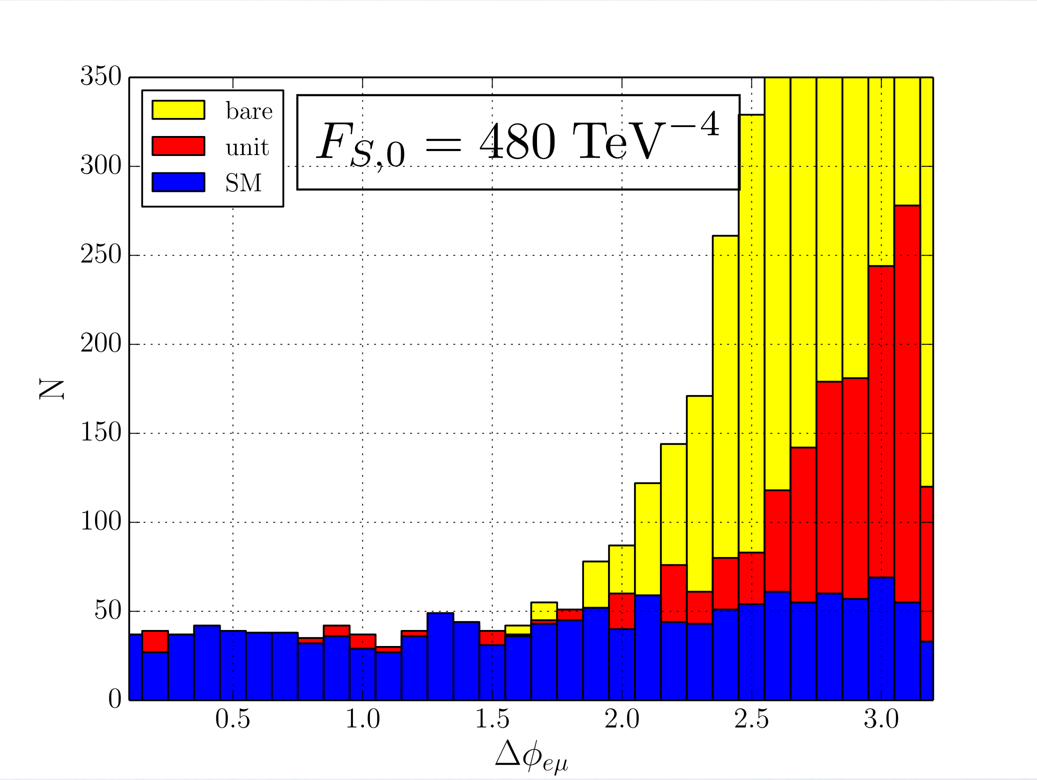


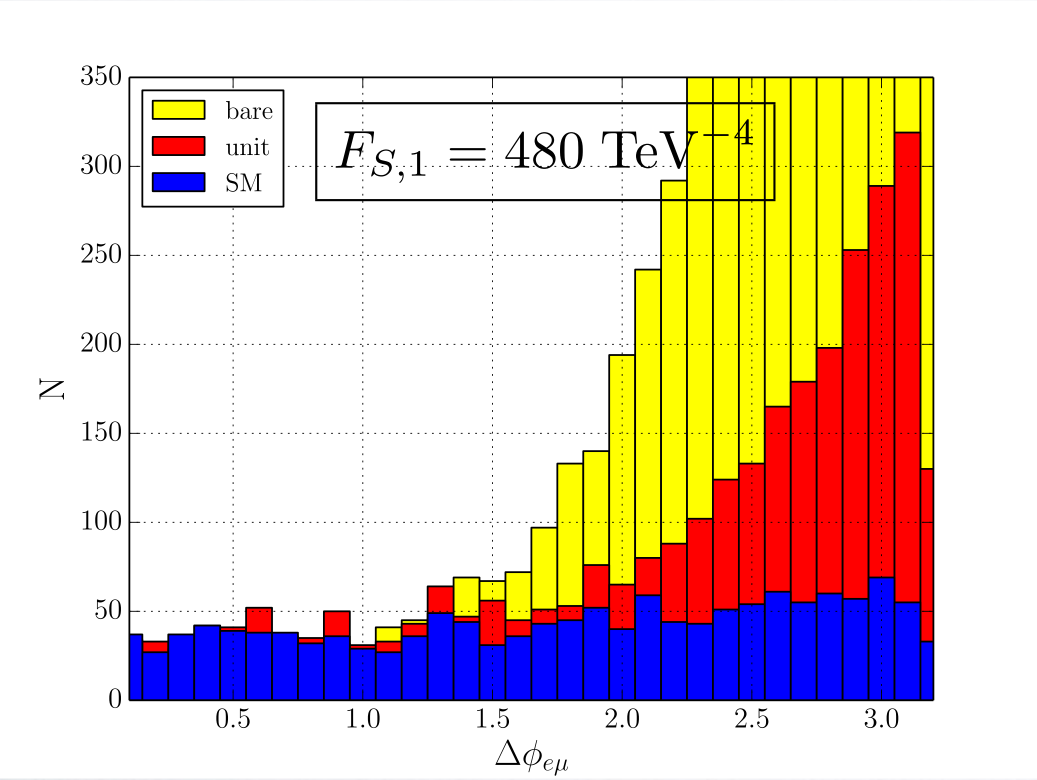


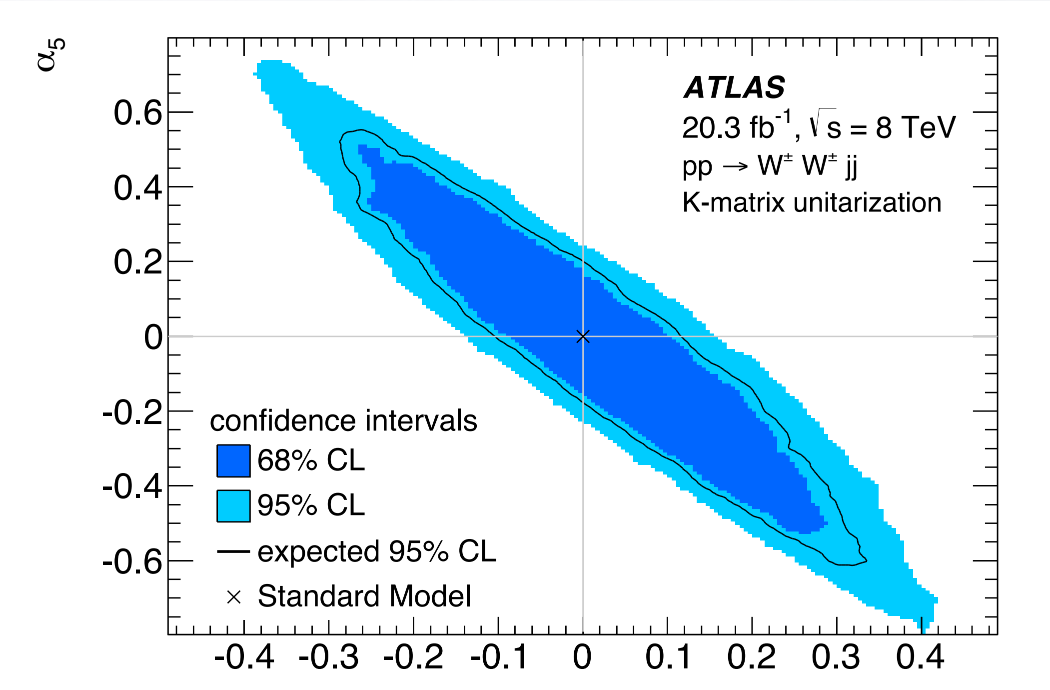


New Physics in Vector Boson Scattering







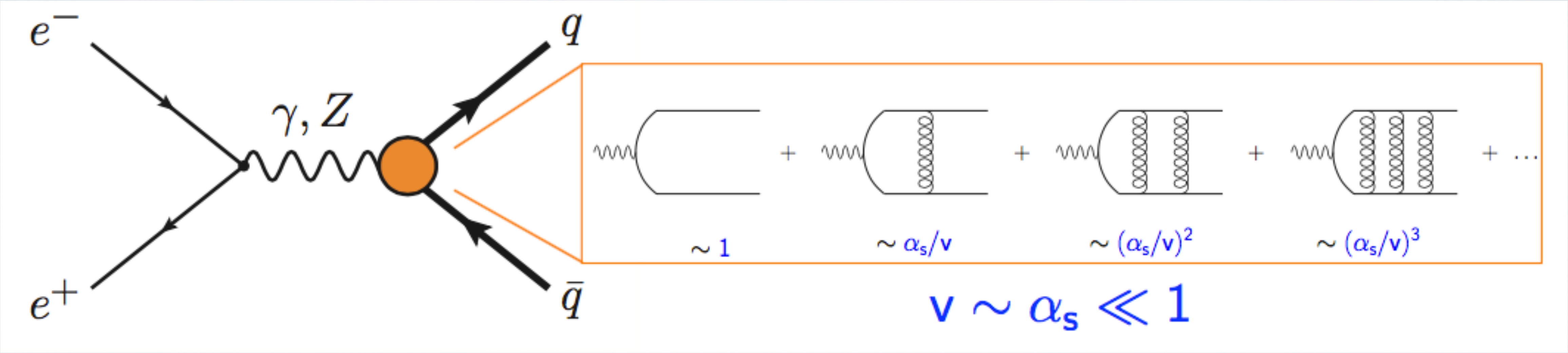


ATLAS, PRL 2014

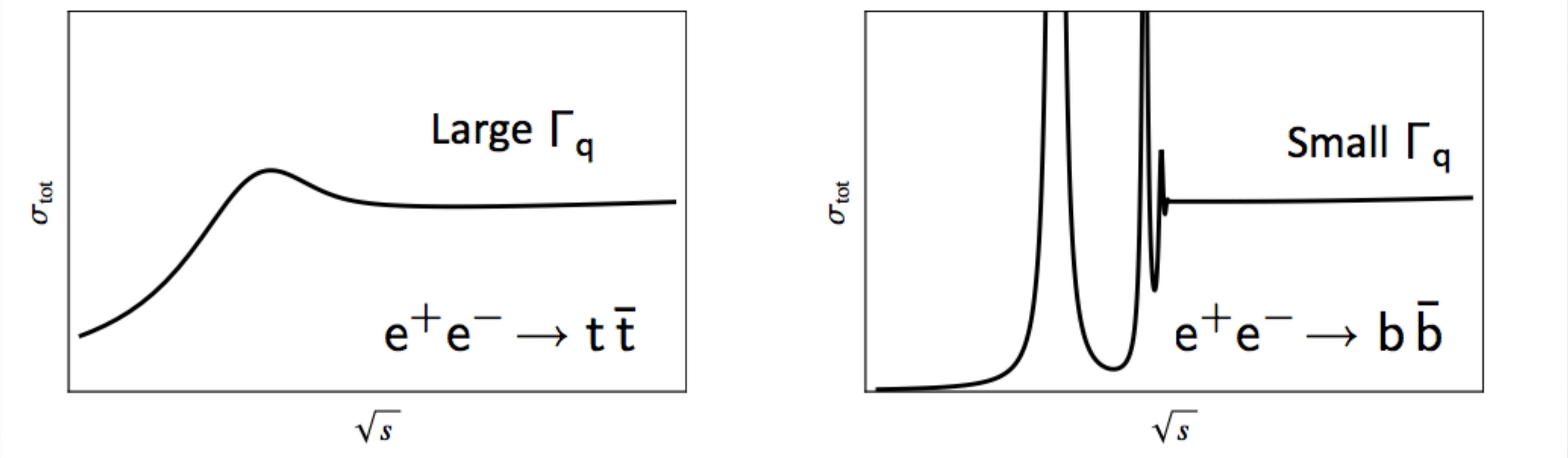
Top Threshold at lepton colliders

ILC top threshold scan best-known method to measure top quark mass, $\Delta M \sim$ 30-50 MeV

Heavy quark production at lepton colliders



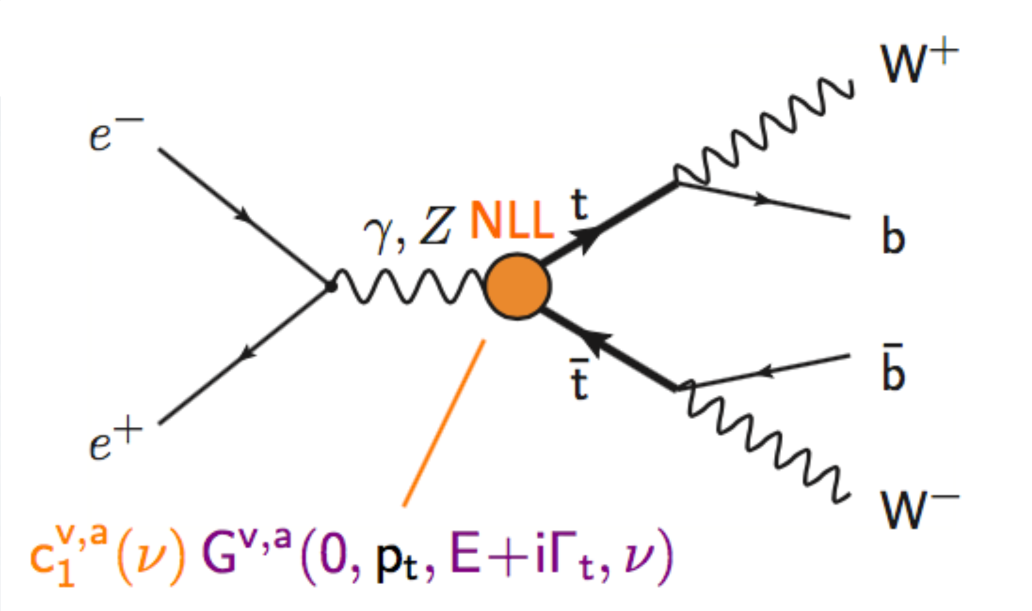
Threshold region (quantitatively)

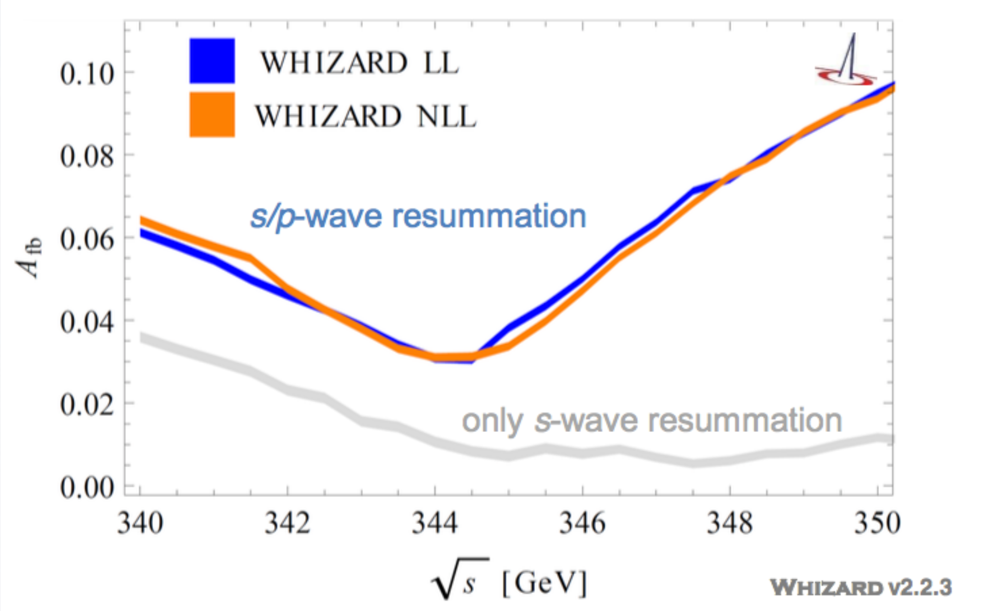


Top Threshold in WHIZARD

with F. Bach/A. Hoang/M. Stahlhofen

- Implement resummed threshold effects as effective tab vertex [form factor] in `WHIZARD`
- $G^{v,a}(0, p_t, E + i\Gamma_t, \nu)$ from TOPPIK code [Jezabek/Teubner], included in `WHIZARD`

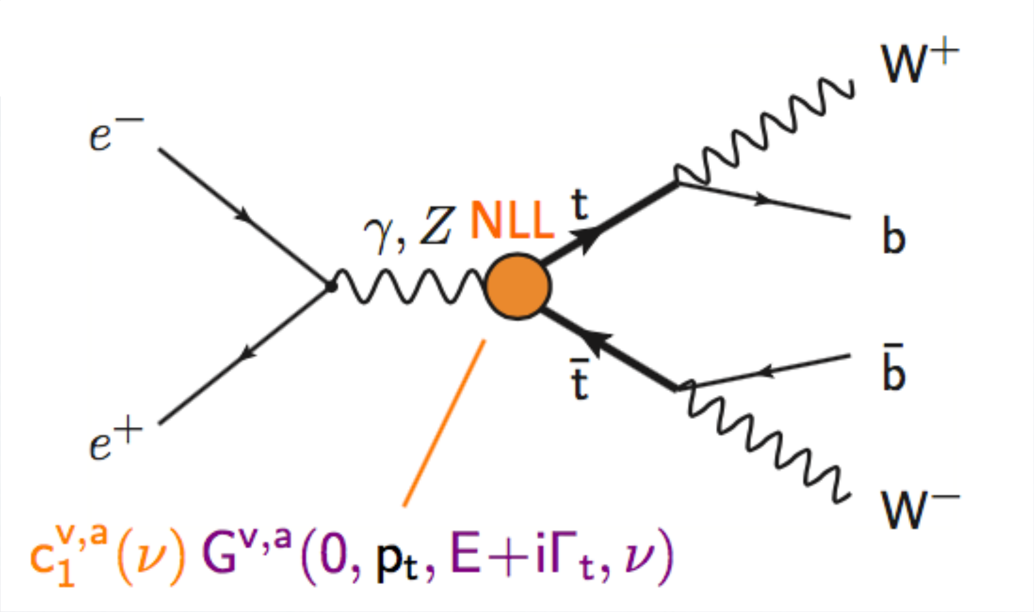


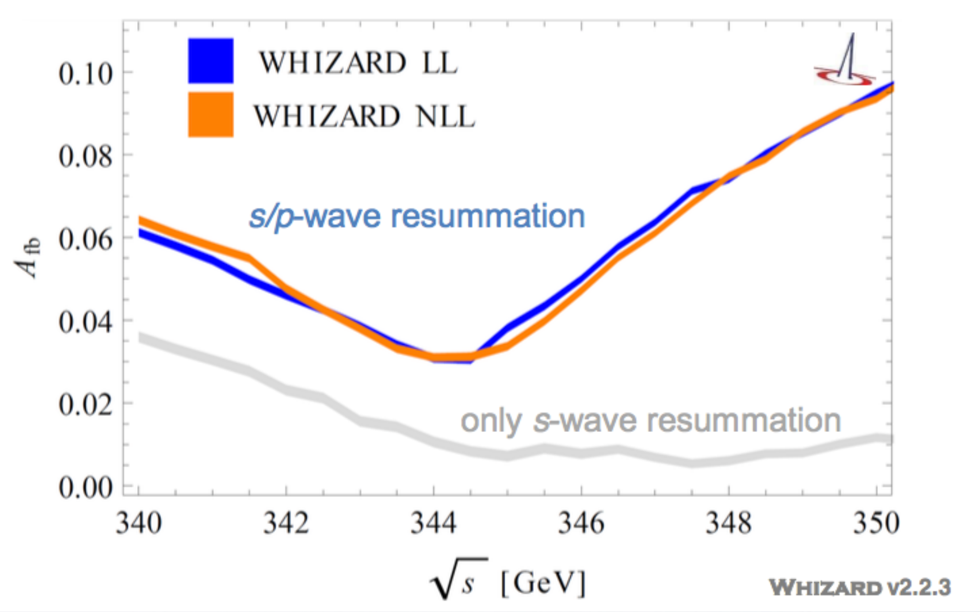


Top Threshold in WHIZARD

with F. Bach/A. Hoang/M. Stahlhofen

- Implement resummed threshold effects as effective tab vertex [form factor] in `WHIZARD`
- $G^{v,a}(0, p_t, E + i\Gamma_t, \nu)$ from TOPPIK code [Jezabek/Teubner], included in `WHIZARD`





$$R^{\gamma,Z}(s) = \underbrace{F^v(s)R^v(s)}_{\text{s-wave: LL+NLL}} + \underbrace{F^a(s)R^a(s)}_{\text{p-wave}\sim v^2\text{:NNLL}}$$

BUT: differentially p-wave at NLL !

- Default parameters:

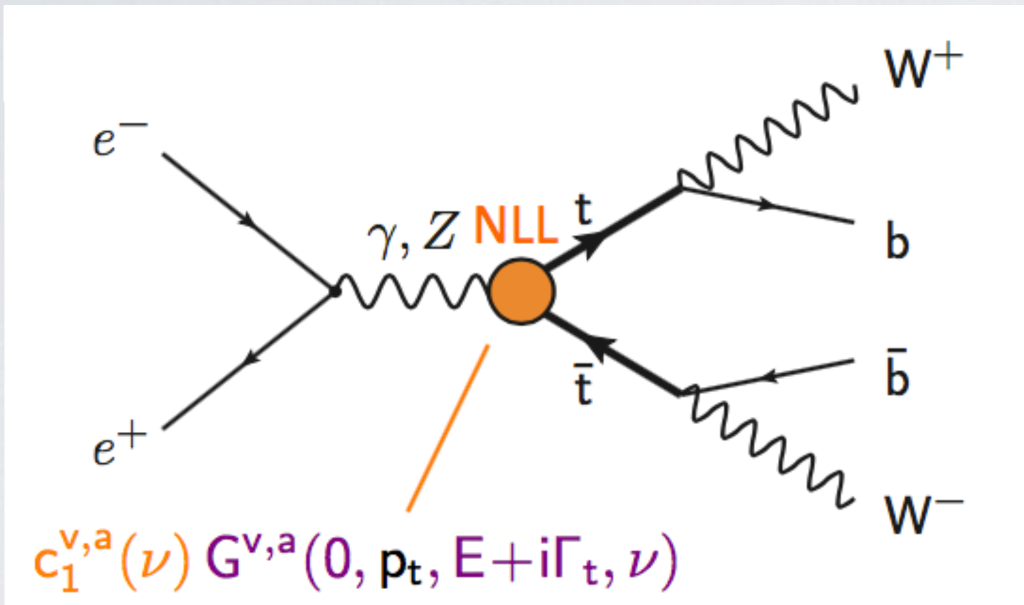
$$\mathbf{M^{1S} = 172\,GeV},\ \Gamma_t = 1.54\,\text{GeV}, \quad \alpha_s(M_Z) = 0.118$$

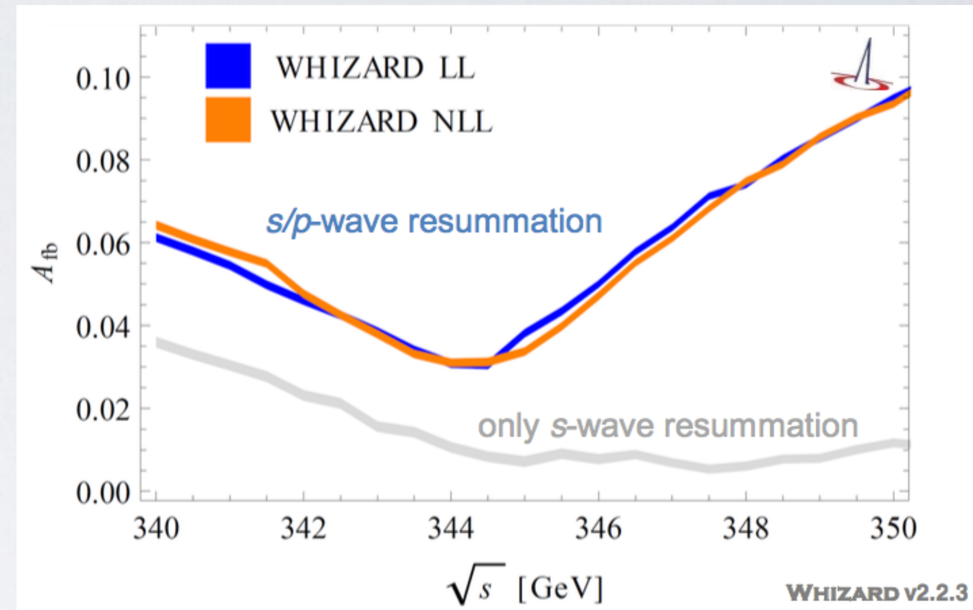
Threshold/Continuum Matching: WIP

Top Threshold in WHIZARD

with F. Bach/A. Hoang/M. Stahlhofen

- Implement resummed threshold effects as effective tab vertex [form factor] in WHIZARD
- $G^{v,a}(0, p_t, E + i\Gamma_t, \nu)$ from TOPPIK code [Jezabek/Teubner], included in WHIZARD





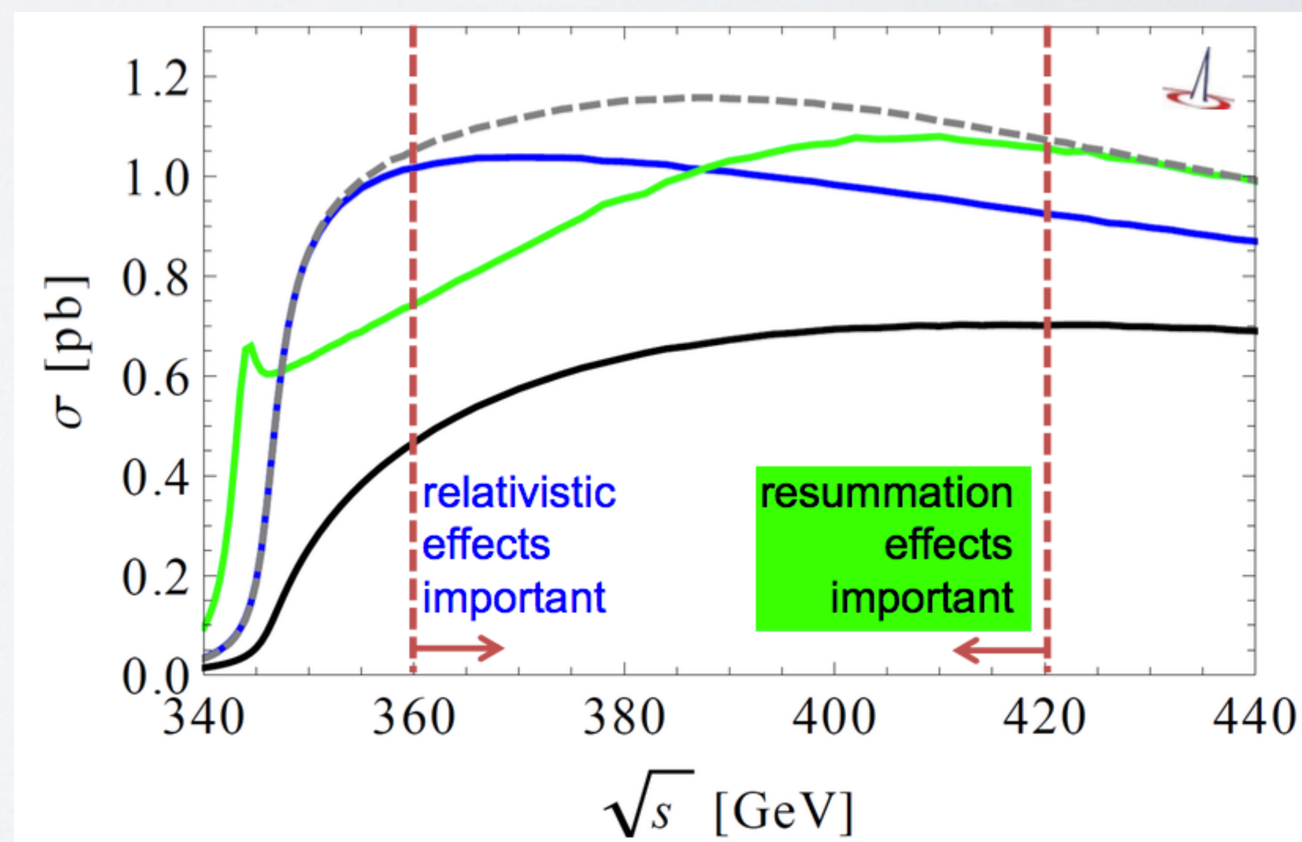
$$R^{\gamma,Z}(s) = \underbrace{F^v(s)R^v(s)}_{\text{s-wave: LL+NLL}} + \underbrace{F^a(s)R^a(s)}_{\text{p-wave}\sim v^2\text{:NNLL}}$$

BUT: differentially p-wave at NLL !

- Default parameters:

$$\mathbf{M^{1S} = 172\,GeV},\ \Gamma_t = 1.54\,\text{GeV}, \quad \alpha_s(M_Z) = 0.118$$

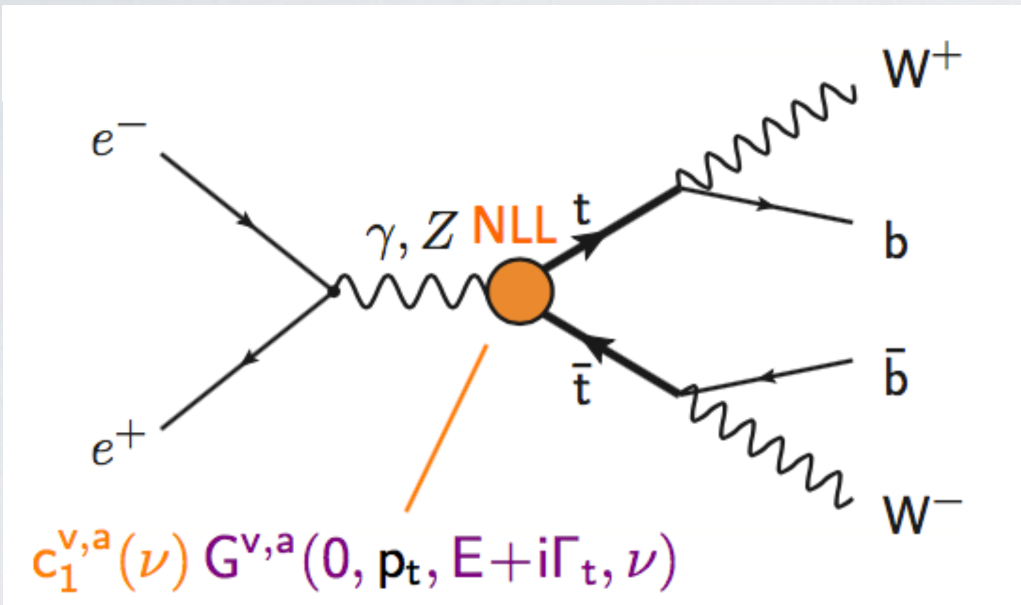
Threshold/Continuum Matching: WIP

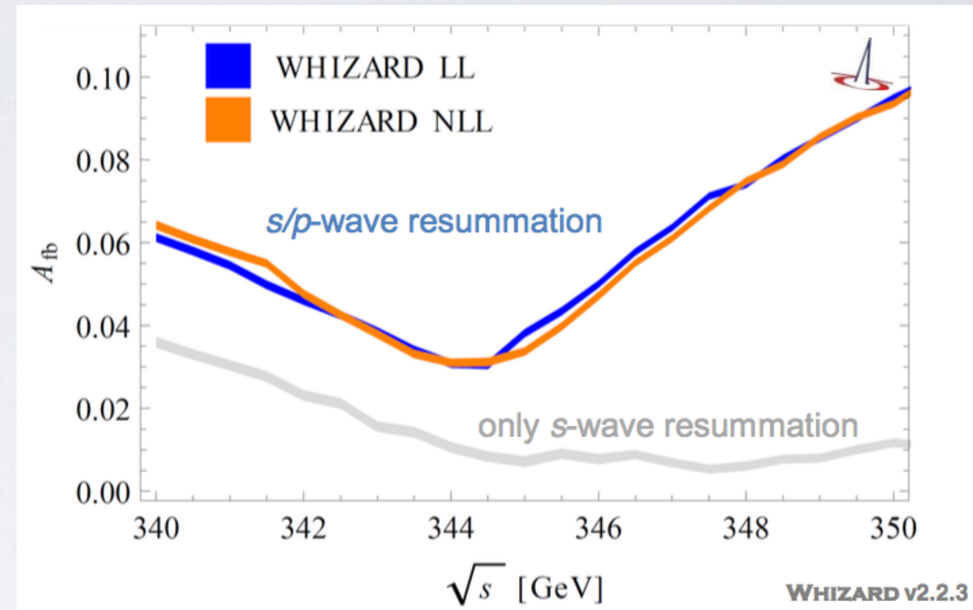


Top Threshold in WHIZARD

with F. Bach/A. Hoang/M. Stahlhofen

- Implement resummed threshold effects as effective tab vertex [form factor] in `WHIZARD`
- $G^{v,a}(0, p_t, E + i\Gamma_t, \nu)$ from TOPPIK code [Jezabek/Teubner], included in `WHIZARD`





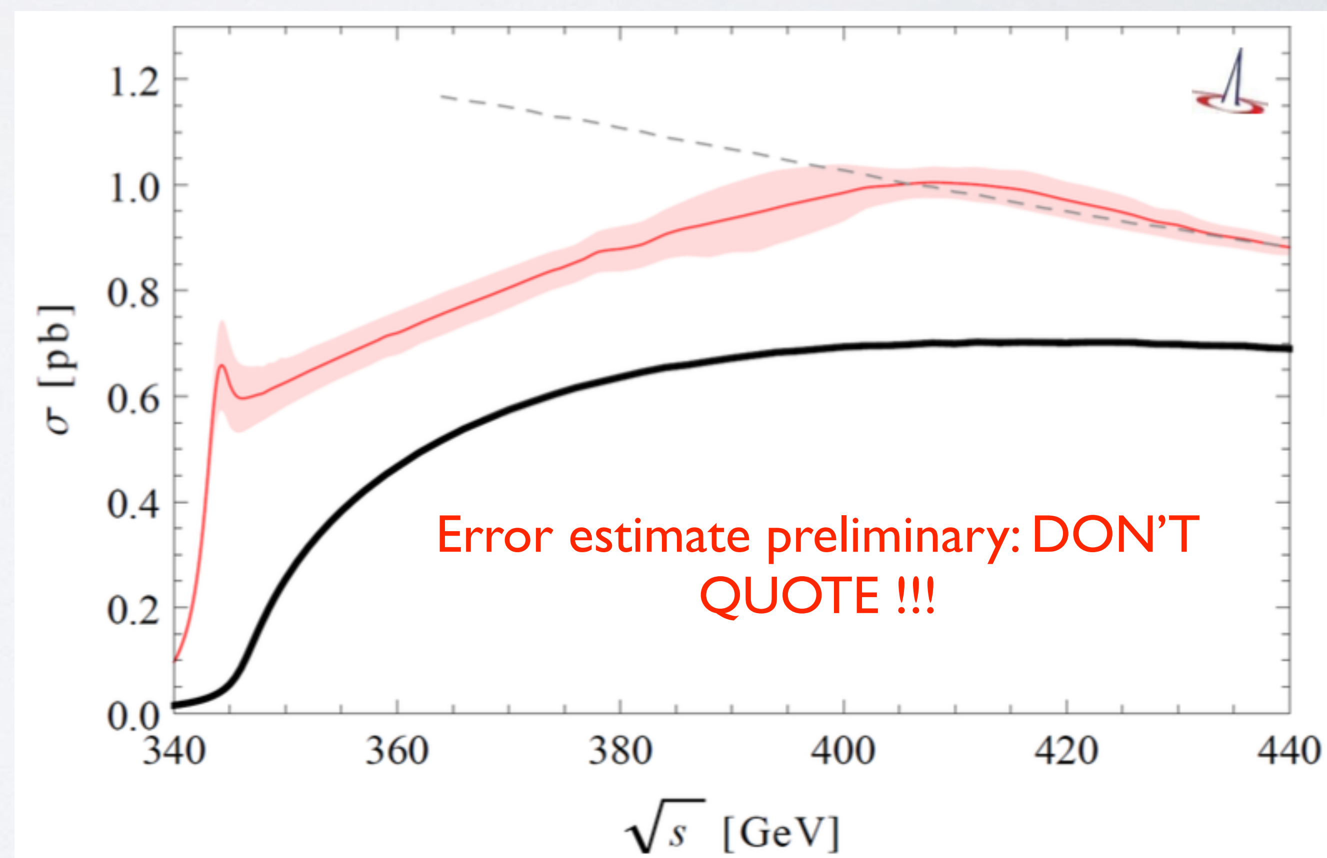
$$R^{\gamma,Z}(s) = \underbrace{F^v(s)R^v(s)}_{\text{s-wave: LL+NLL}} + \underbrace{F^a(s)R^a(s)}_{\text{p-wave}\sim v^2\text{:NNLL}}$$

BUT: differentially p-wave at NLL !

- Default parameters:

$$\mathbf{M^{1S} = 172\,GeV},\ \Gamma_t = 1.54\,\text{GeV}, \quad \alpha_s(M_Z) = 0.118$$

Threshold/Continuum Matching: WIP



Projects, Plans, Performance and all that

- O'Mega Virtual Machine (OVM): ME via bytecode interpreter than compiled code ✓
- Parton shower: LO merging (MLM ✓) , NLO matching
- QED shower (FSR)
- QED shower (ISR); exclusive part of ISR spectrum
- pT spectrum of ISR radiation
- automated massless/massive QCD NLO corrections: FS ✓ / Initial state in preparation
  → `WHIZARD 3.0`
- QED/electroweak NLO automation: longer time scale
- complete NLL NRQCD top threshold/NLO continuum matching; extension to ttH [✓]
- POWHEG matching implemented ✓ ; maybe also MC@NLO or Nagy-Soper matching
- Monte Carlo over helicities and colors
- Modified algorithm for multi-leg (tree) matrix elements: includes high-color flow amplitudes, QCD/EW coupling orders, general Lorentz structures
- Automatic generation of decays (and calculation of decay widths) ✓
- New syntax for nested decay chains

```
process = e1, E1 => (t => (Wp => E2, nu2), b), tbar
```

Conclusions & Outlook

- `WHIZARD 2.2` excellent tool for Collider Physics (ee, pp, ep)
- Allows to simulate all possible BSM models
- Allows for all SM backgrounds
- ee physics: beamspectra, LCIO, LC top threshold
- Main focus in physics: NLO automation $\longrightarrow$ `WHIZARD 3.0`
- Ongoing projects: Lorentz structure, showers, merging
- Tell us what is missing, insufficient, annoying, desirable

Conclusions & Outlook

- WHIZARD 2.2 excellent tool for Collider Physics (ee, pp, ep)
- Allows to simulate all possible BSM models
- Allows for all SM backgrounds
- ee physics: beamspectra, LCIO, LC top threshold
- Main focus in physics: NLO automation $\longrightarrow$ WHIZARD 3.0
- Ongoing projects: Lorentz structure, showers, merging
- Tell us what is missing, insufficient, annoying, desirable

even if it is in a conference summary talk $\Longrightarrow$ Challenge accepted !

Conclusions & Outlook

- `WHIZARD 2.2` excellent tool for Collider Physics (ee, pp, ep)
- Allows to simulate all possible BSM models
- Allows for all SM backgrounds
- ee physics: beamspectra, LCIO, LC top threshold
- Main focus in physics: NLO automation $\rightarrow$ `WHIZARD 3.0`
- Ongoing projects: Lorentz structure, showers, merging
- Tell us what is missing, insufficient, annoying, desirable

even if it is in a conference summary talk $\Longrightarrow$ Challenge accepted !

One Ring to Find them … or One Ring to Rule them out

One Ring to Find them … or One Ring to Rule them out