News from WHIZARD

Wolfgang Kilian

University of Siegen

LCWS, Strasbourg
October 2017

WHIZARD: Overview

Scope

WHIZARD is a stand-alone program for processes at high-energy colliders: scattering and particle decays

- integrate cross sections and decay widths (perturbative, partonic)
- account for LC beam structure
- amplitude code is generated and executed on the fly
- shower/hadrons via internal or external code
- calculate observables, generate event samples

Universal Monte Carlo for elementary processes at (future lepton) colliders

WHIZARD for LC Studies

SM event samples (DBD) for LC studies ⇒ T. Barklow

- WHIZARD 1
- Leading-order (QCD and EW), PYTHIA 6 for shower/hadrons
- Various adjustments and add-ons for specific LC issues

Present and future studies: WHIZARD 2

- user interface (scripting language SINDARIN)
- internals redone aimed at NLO and modularity
- CIRCE 2 for detailed beamstrahlung spectra
- current activity: validation, refinements, NLO (QCD and EW)
- HPC parallel evaluation
- New simulations/samples for ILC/CLIC studies (LC generator group)

Team

- Program exists since 1999
- Active authors and developers:
  - WK (U Siegen), T. Ohl (Würzburg), J. Reuter (DESY)
  - S. Brass, V. Rothe, M. Sekulla, C. Schwinn, S. Shim, F. Staub, P. Stienemeier, M. Utsch, Z. Zhao
- Upcoming version:
  - 2.6.1 (3. Nov. 2017)

For the User

- Installation centrally or locally on any Linux or Mac (`autotools`)

```
./configure [options]
make
make install
```

- Look-and-feel like standard Linux/Unix apps

```
./whizard [options] FILE
```

- User works in directories at his choice, no structure imposed

Programming Languages

- Main program code is written in modern Fortran (F2008).
  Compilers: `gfortran`, `ifort`, `nagfor`
- Glue code uses `Make`, Makefiles generated on-the-fly
- Algebra is done in `OCaml` (compiler free for all platforms)
  ⇒ Sub-package `OMega`
- Some interface to external code in `C/C++`
- Can use precompiled libraries (`LHAPDF`, `HepMC`, . . .)

Input

- Models precompiled or generated (FeynRules, SARAH, UFO)
- User provides all other information via input file, which is actually a script
  - Model selection, data, parameters, process setup
  - Compute observables for in/out particles: cuts, weight for integration
  - Integrate once, or loop and scan over parameters
  - Generate and write event samples, select format
  - Analyze event samples and plot results

  Programming language (DSL): Sindarin

Output

1. Numbers (cross sections, decay widths, other observables)
2. Distributions (plots)
3. Event samples in various formats
   (fixed-order or showered, partonic or hadronic)

Variants

- Docker package (VM) with all prerequisites
- Library version (C-compatible API) for inclusion in other programs
- Interactive mode for simple tasks
- GUI (`Javascript/node.js`) as server/client model version via Web browser

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 couplings	SM_ac_CKM	SM_ac
SM with anomalous top couplings	SMtop_CKM	SMtop
SM for e^+e^- top threshold	–	SM_tt_threshold
SM ext. for VV scattering	—	SSC / SSC2
SM ext. for unitarity limits in VV	—	SM_ul
SM with Higgs singlet	—	HSExt
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

Recent additions to the Model list

- EFT for the SM: complete bosonic $D = 6$ Lagrangian (Warsaw Basis)

  S. Shim

- Beyond EFT: electroweak interactions at high energy
  $\Rightarrow$ (Super)LHC/FCC-pp, ILC and CLIC

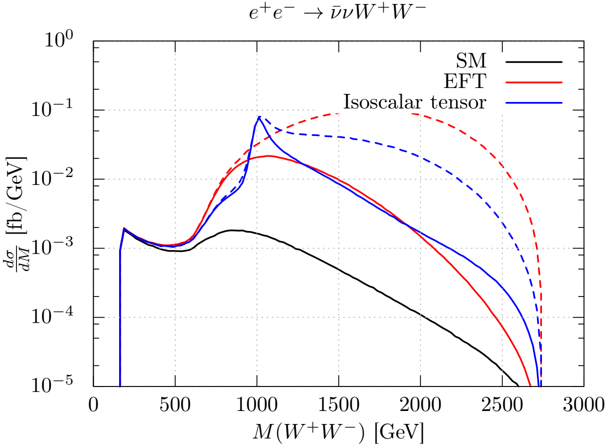
  C. Fleper, M. Sekulla

- UFO File Support (`OMega`)

  T. Ohl

CLIC @ 3.0 TeV: Resonance with $M = 1.0$ TeV

Simplified Model



Collider Setup

Trivial beam structure: just process definition

```
process foo = "e+", "e-" ⇒ "mu+", "mu-"
```

⇒ scattering process or decay process

Nontrivial beam structure:
polarization, momentum, angle, beam spectra, structure functions

e^+e^- Collider Beam Structure

1. Beamstrahlung = classical beam-beam interaction, collective effect
2. ISR = resummed soft/collinear QED bremsstrahlung for initial-state electron-electron interaction
3. EPA = $\gamma\gamma$-induced processes

e^+e^- Collider: Beamstrahlung

Options for handling beamstrahlung (as supported by WHIZARD)

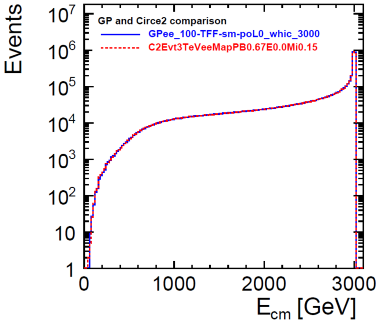
1. Beam-energy spread
   $\Rightarrow$ Gaussian distribution

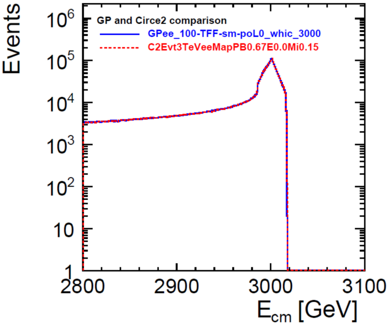
2. `CIRCE1`: parameterized beam spectrum
   $\Rightarrow$ Factorized, few parameters in fit

3. Beam-events file: use result of GuineaPig simulation directly
   $\Rightarrow$ Finite number of pre-simulated beam events

4. `CIRCE2`: generator takes binned and interpolated result of detailed GuineaPig simulation
   $\Rightarrow$ Most precise descriptionh

e e Beam Spectrum 3TeV





Left: dN/d√s distribution Guineapig events (blue) and Circe2 generated events using a power function mapping (red); 20 GeV bins.
Right: zoom into the high √s region; 1 GeV bins; looks good.
Check the distribution ratios.

July 2016 J-J.Blaising, LAPP/IN2P3 4

Processes

```
process foobar =
  "e+", "e-" => nue, nuebar, b, bbar, q, qbar, "mu+", "mu-"
```

- Explicit, exclusive
- optionally constrained
- optionally with sum over flavors `alias q = u:d:s:c`
- optionally sum over subprocesses ⇒ inclusive, e.g. decays

Amplitudes: `OMega` (automatic recursive tree-level amplitudes)

Processes at NLO

- NLO QCD: implementation complete
  - Interfacing GoSam, OpenLoops, Recola
  - FKS subtraction
  - $t\bar{t}$ and $t\bar{t}H$ off-shell: $\Rightarrow$ J. Reuter's talk
  - currently validating large class of NLO processes
  - POWHEG matching for e^+e^-
- NLO SM: complex mass scheme
  - Interfacing GoSam, OpenLoops, Recola
  - FKS subtraction / bookkeeping for photons w.i.p.

Phase Space and Integration

Phase Space

Multi-Channel: Select dominant singularity structures, parameterize via resonance mappings etc.

(Selection is automatic, heuristics for determining channel importance)

Integration

Twofold adaptive: VAMP

1. Each channel gets a weight which is iteratively adapted
2. Each channel is binned (VEGAS-like), binning is iteratively adapted

Warmup iterations yield integration grid and channel-weight distribution optimized for the particular process, to be used in event generation

Parallel Processing

OpenMP

Make use of multi-core processors: parallel evaluation of helicity configurations

MPI

Make use of multi-processor clusters: parallel sampling of phase space
MPI 3.0 asynchronous message passing
$\Rightarrow$ New feature S. Brass

Issues of parallel processing

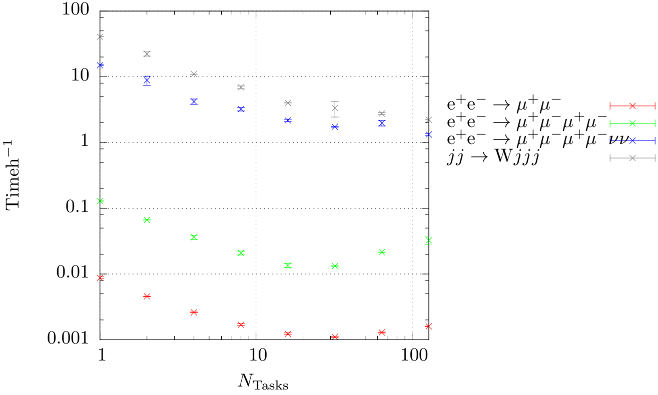
- Parallel speedup can be perfect if all processors work on independent data and don't combine their results
  - Event generation
  - Integration between adaptation steps
- Limits arise if bunches of data have to be exchanged.
  - File I/O
  - RNG state
  - Grid updates.
- Limits arise if parts of the program runs serially
  - Initial setup, Sindarin reader, matrix-element generation, phase-space construction, . . .

New solutions for parallel processing (WHIZARD 2.6)

- Pure calculations for event generation and integration now run trivially in parallel, as required (MPI/HPC cluster)
- Grid construction and exchange: communication
  Now use MPI 3.1: asynchronous communication, no blocking
- RNG state: statistically independent evaluation, but reproducible
  Now use a modern stream RNG with block-skip functionality
- Serial parts: currently the limiting factor
  MPI 3.1: run serial part only on one image; automatic switch-on/off of needed worker nodes
  Soon: Much faster phase-space construction; planned: parallel construction

S. Brass, M. Utsch

Speedup (WHIZARD 2.6.0 with MPI)



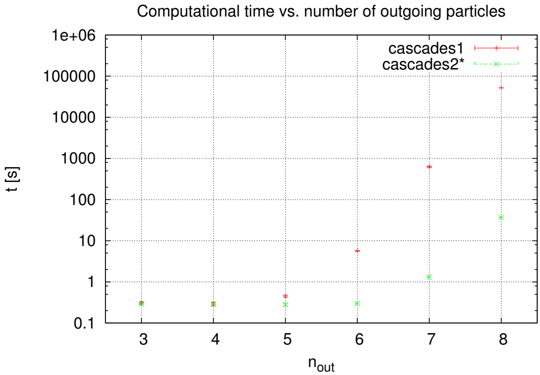
Speedup by factor > 10, currently saturates at $O(10\ldots100)$ workers

Bottleneck of this strategy: Phase-space construction

Fast phase space construction (WHIZARD 2.6.1+)

Use O'Mega for phase-space graph structure (DAG exists already)

$gg \to W^+ b\bar{t} + n\gamma$



M. Utsch

MPI parallel version ⇒ further speedup possible

Event Handlers

WHIZARD uses the twofold-adapted phase space to generate unweighted event samples. Further processing:

1. Particle decays (cascades):
   - using WHIZARD's own decay processes (explicit or automatic)
   - full control over polarization transfer (uncorrelated, classical, quantum correlation)
2. Photon radiation: exclusive photons from inclusive ISR
3. Resonance histories: control shower behavior
4. Shower, Hadronization: optionally call PYTHIA internally
5. Event output: file formats StdHEP, LHEF, HepMC, LCIO, ASCII

Photon Handler

Inclusive ISR description (LL soft, 3rd order collinear) accounts for precise cross section and energy dependence

Caveat: all radiated energy strictly collinear.

Ad-hoc solutions for p_T spectra in WHIZARD 1 and WHIZARD 2 had deficiencies (and were mutually incompatible).

New Photon Handler in WHIZARD 2.6

Take generated events

- Collinear photons (both beams) are given transverse momentum according to scale-less logarithmic distribution (w/ cutoffs)
- Both beams handled: exact energy-momentum conservation
- Common scheme for ISR (e^+e^- process) and EPA ($\gamma\gamma$ process)

Plan: extend to multiple photons & match with NLO

Resonances and Parton Shower

Current standard for simulation with WHIZARD 2

PYTHIA 6 parton shower (internal). Ongoing validation by LC generator group.

Important LC processes: $(WW, ZZ, ZH \to)4f$, $(WWZ, ZZZ, WWH, \ldots \to)6f$, etc.: contain resonances

PYTHIA 6 modes:

1. Default: local interaction, shower starts at process energy scale, invariant masses reshuffled
2. Resonance: nonlocal interaction, shower starts at resonance mass, invariant masses fixed

Interplay of resonant/nonresonant background?

Solution for DBD (WHIZARD 1): T. Barklow

Determine resonances as color singlets $\Rightarrow$ retain invariant mass

- Works for SM W and Z (and H)
- Ignores non-resonant background, subleading $1/N_c$ and color reconnection
- Precise simulations for ILC?
- Overlapping resonances ($H^{(*)} \to WW^{(*)}$)?

Preferred solution for future event samples

1. Determine momentum reshuffling / high-pT radiation by explicit calculation
2. merge with no-radiation + low-pT shower sample
3. Consistent NLO/NLL simulation ... likely still insufficient

New solution for WHIZARD 2 (WIP)

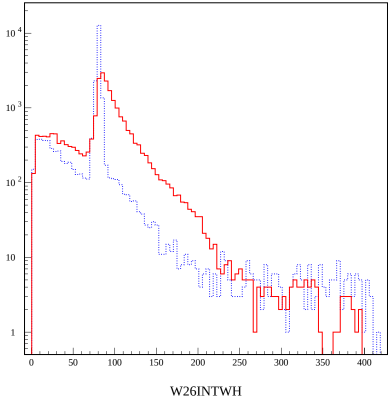
Generate event sample as usual, no change to cross section (LO).

1. Determine possible resonance histories for each single event
2. For all applicable histories, compute factorized matrix elements *in addition to* complete matrix element
3. Use relative ME values to determine probabilities for histories, including background (remainder)
4. Select one of the applicable histories to modify the event record

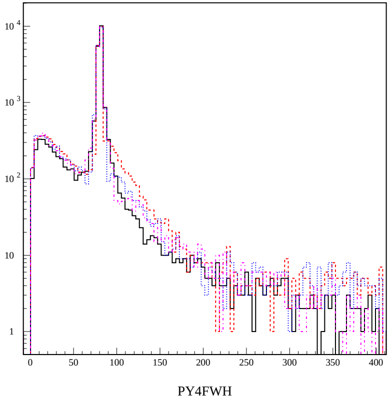
Necessary: cutoff (how far off shell is still on shell?)

Resonance / Background in Event History

Resonance by color / no resonance



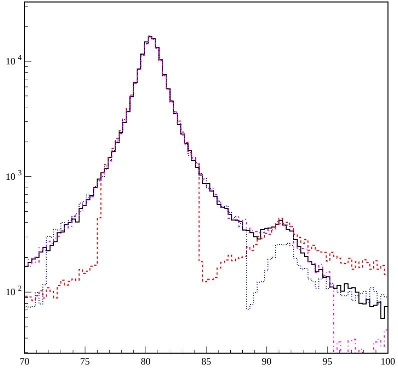
Resonance by ME, varied cutoff



Plots by M. Berggren (WIP)

Cutoff Artefacts?

Resonance by ME, varied cutoff



⇒ insert smooth transition

Plots by M. Berggren (WIP)

Summary: Recent Developments in WHIZARD

- NLO QCD (under validation)
- UFO support & nonperturbative electroweak (simplified) models
- Parallel evaluation with MPI
- Photon p_T distribution in ISR/EPA event samples
- Interplay of resonances with parton shower

whizard.hepforge.org