Whizard requests from CLICdp

Ulrike Schnoor, Philipp Roloff

4-6 November 2019
Mini-workshop for the LC generator working group

Outline

Radiative Bhabha scattering

Alternative weights

MPI for processes with resonance history

Integration grids with CIRCE2

Displaced decay vertices in Whizard

Radiative Bhabha scattering

Launchpad https://answers.launchpad.net/whizard/+question/685180

Status Open

Details

- Process to generate: $e^+e^- \to e^+e^-\gamma$ (background to monophoton if both of the electrons are either very soft or stay in the beam pipe)
- ideally no cuts can be placed on the electrons, only on the photon
- Requirements on the photon do also bias the electron kinematics a bit, but nevertheless we do not get a convergence with Whizard if no electron cuts are applied (this is expected because there are singularities which are only cancelled at higher order, which are not present in the generation)
- How can we know in a well-defined way what is the part of the cross section we are missing when we apply cuts on the electrons?
- Can we generate a sample with a cut on the electron theta and energy that removes the singularity and still be somewhat sure that we are not missing too much of a contribution?
- Is there a way to get the higher order in Whizard or do you know some other tool that can do it?

Sindarin An example sindarin can be found here:
https://cernbox.cern.ch/index.php/s/1i3EFSz0BGS6ftS

More info More info https://indico.cern.ch/event/842877/contributions/3602728/

Alternative weights

Goal TGC samples (WW → qqlnu) with production based on the reweighting procedure described here: `http://bib-pubdb1.desy.de/record/94888`

Problem Propagating the weights for each event based on alt_setup

Launchpad `https://answers.launchpad.net/whizard/+question/684019`

Status LCIO alternative weights are written: SOLVED in Whizard 2.8.2 - Thank you!

New problem:

- Alternate weights are all (except for numerical precision) = 1: for `g1z=1.05, ka=1.05, la=0.05`

	σ [fb]
SM	380
TGC point direct	902
TGC alt_setup	380

(direct: not using alt_setup, modifying the couplings parameters directly)

The cross section is a factor of 2.4 higher, but the alternative weights are all basically = 1

- Can alt_setup be used for TGC at all? What would be another option (except generating a sample for each TGC point separately)?
- Sindarin file: `https://cernbox.cern.ch/index.php/s/ibsvYyAi8AmW51q`
- Is it correct to use the following settings?
  ?unweighted = true (tried false: still no difference between the alternative and nominal weights [also: LCIO events must be unweighted])
  ?update_sqme = true
  ?update_weight = true

MPI for processes with resonance history

Problem MPI running does not work with resonance history switched on

Details Whizard gets stuck at "MPI: wait for master to finish process initialization"

This is actually also mentioned in the manual (3.3.2) that for processes with resonance history the parallelization with MPI is not yet available.

Status ?

Integration grids with CIRCE2

Problem Whenever beam spectra (circe2) are used, the integration grids are discarded when running in the same directory:
`VAMP: parameter mismatch, discarding grid file 'decay_proc.m1.vg'`
although it works when there is no circe2 used

E-Mails from March 2019

Status Open, but existing workaround

Workaround `?check_grid_file = false`

Displaced decay vertices in Whizard

Problem For Long-Lived Particles (LLP) searches, need displaced decay vertex in the event record

Details

- Process: chargino pair production where the charginos decay to a neutralino and a pion: $e^+e^- \to \tilde{\chi}_1^+\tilde{\chi}_1^- \to \tilde{\chi}_1^0\pi^+\tilde{\chi}_1^0\pi^-$
- Chargino mass $m_{\tilde{\chi}_1^\pm} = 1050\,\text{GeV}$ (PDGID = 1000024),
  neutralino mass $m_{\tilde{\chi}_1^0} = 1049.8\,\text{GeV}$ (PDGID = 1000022)
- Chargino lifetime $c\tau = 6.9\,\text{mm}$ (to be varied)
- WHIZARD-2.7.0 used for generation of the hard process, PYTHIA6 for the parton shower
  - Full chain in WHIZARD up to neutralino and pion (u,d) final state, with PYTHIA used for parton shower,
    OR:
  - PYTHIA can also do the chargino decay, passing the relevant parameters to the WHIZARD-PYTHIA interface: `$ps_PYTHIA_PYGIVE = "IMSS(1)=1;PMAS(312,1)=1050.; PMAS(312,4)=60.;MDCY(312,2)=2601;MDCY(312,3)=1;PMAS(310,1)=1049.8;MDCY(310,1)=0; KFDP(2601,1)=1000022;KFDP(2601,2)=211;KFDP(2601,3)=0;KFDP(2601,4)=0;KFDP(2601,5)=0;BRAT(2`
  - Using WHIZARD for the full chain has the advantage that neutralino event records can be handled as stable particles by Geant
- Event record from WHIZARD does not contain the displacement

Status Open, but existing workaround

Workaround Using Geant4 to propagate the LLP (see next slide)

[FYI] Workaround: Lifetime in Geant

Workaround: Use GEANT4 to obtain the displaced decay by setting the chargino lifetime in the GEANT4 particle table[1]

- In `ddsim`, this is done via the option `--physics.pdgfile particle.tbl`
- New function in ddsim: `setExtraParticles(extraParticles)` where `extraParticles` is of the form [ID, name, chg, mass, total width, lifetime]
  or
  "ID name chg mass total-width lifetime"
- In the production system, use ddsim `ExtraParticles="..."` setting
- link to iLCDiracDoc

[1] `https://github.com/AIDASoft/DD4hep/blob/master/DDG4/examples/particle.tbl`

[FYI] To reproduce: Process setup in Whizard

In the Sindarin file, put

- correct masses of chargino, neutralino
- correct width of the chargino in GeV (currently little to no influence since the displaced decay is not written)
- correct chargino mixing matrix
- set the masses of other susy particles to high values
- processes:

```
process charginopair = e1, E1 => "ch1+", "ch1-"
process dec_chargino1m  = "ch1-" => ubar, d, neu1
process dec_chargino1p  = "ch1+" => dbar, u, neu1
unstable "ch1-" (dec_chargino1m)
unstable "ch1+" (dec_chargino1p)
```

- beams, circe2, polarisation, isr handler, parton shower
- integration and simulation:

```
integrate (dec_chargino1m, dec_chargino1p) { iterations = 1:1000 }
integrate (charginopair) { iterations = 5:10000, 2:10000 }
simulate (charginopair) {}
```