GUINEA-PIG vs. GUINEA-PIG++

Dominik Aromiński

Acknowledgements:
D. Schulte, B. Dalena, A. Latina, A. Sailer

CLIC beam physics meeting

21.12.2017

Overview

1. Introduction
   - I/O differences
2. Guinea-Pig benchmarking
   - Luminosity and background yields at 3 TeV example
3. Summary and outlook

Introduction

- Guinea-Pig simulates the beams collisions and background creation & provides luminosity spectra for machine tuning
- Original manual available in Daniel's thesis
- The most recent report on Guinea-Pig available in Daniel's presentation
- Guinea-Pig++ description is available in LAL webpages: official documentation webpage (partially outdated) or Guinea-Pig++ report

Why move to Guinea-Pig++?

- It contains everything Guinea-Pig has and more
- Written modularily in C++ in an object-oriented paradigm
- Code is easier to maintain and add new features to the program; much easier to follow the information flow between objects than in large structural code
- Guinea-Pig++ runs calculations faster ($\approx 20\%$) and is easier to optimize
- C++ became the standard for scientific code; it is more natural and better known to the new generation to physicists and developers

I/O differences

- The beam input files are the same for both codes, and are Placet-compatible if “load_beam = 3;” setting is used
- Some differences appear in additional options in the configuration file acc.dat
- There are some changes in file structure of output in Guinea-Pig++:
  - More robust output file with short descriptions or parameters
  - Incoherent pairs have one additional column with information about the process that lead to its creation
  - Units for beamstrahlung are in μrad instead of radians
- 17 columns in luminosity output files instead of 10; C++ version contains also the information about colliding particles’ momenta and a label

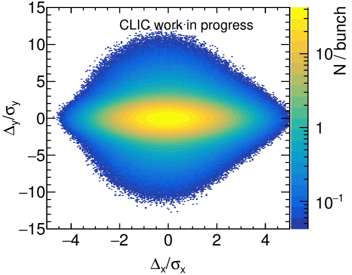
I/O differences c'd

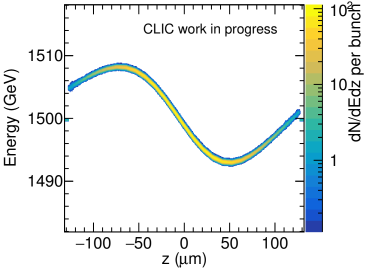
- The name change from "pairs" to "secondaries" as reported in documentation is not valid anymore, the Guinea-Pig-compatible version is the functional argument
- Added possibility to set the polarization vector for the beams
- Added switches for BMT precession and Sokolov-Ternov spin flip
- "Silent" argument does not exist as there is no screen outputting during simulation
- "lumi_p", "lumi_p_eg" and "_gg" arguments have to be specified in Guinea-Pig in order to have luminosity spectra for the other initial states which has the opposite effect in Guinea-Pig++

Guinea-Pig++ new features

- Possible to track and store incoherent muon pairs
- Introduction of trident cascades along with coherent pairs
- Added depolarization due to Sokolov-Ternov spin flip, BMT spin precession
- More advanced treatment of bhabha electrons: boosting and rotating, described thoroughly in documents attached to Guinea-Pig++
- 64-bit random number generator with a possibility of choosing the random seed
- Automatic grid sizing based on beam sizes, offset settings and calculated deflection, though when used by me it seemed to have a substantial amount of particles in beam2 out of the grid
- Alternatively, one can use grid sizing based on beams read from file with "get_cuts_from_loaded_beam" or simply can choose the grid sizes as it was done in Guinea-Pig

Benchmarking assumptions

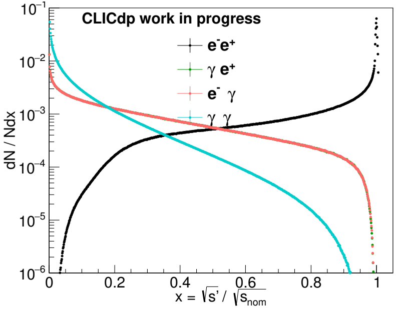




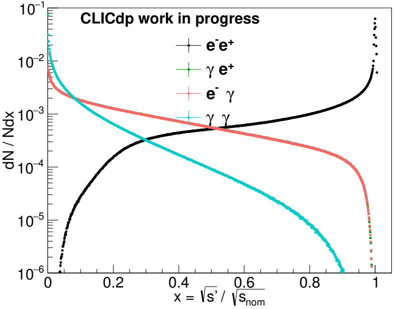
- Newest available releases of Guinea-Pig: 1.4.4 for C-version and 1.2.1 for the C++
- Placet1 tracked the beams through CLIC 3 TeV, L* = 6 m Beam Delivery System with energy spread depending on particle's position in the bunch
- Statistics used: 1 bunch train, ≈ 300 bunch crossings for Guinea-Pig++, and 150 BX with Guinea-Pig
- Grid sizes: cuts at 12 x 64 x 3 sigmas with granularity of 128 x 640 x 25, and a $n_t = 1$

Luminosity spectra in 3 TeV CLIC with L* = 6 m

Guinea-Pig



Guinea-Pig++



design	$\mathcal{L}_{total}$ ($x10^{34}\mathrm{cm}^{-2}\mathrm{s}^{-1}$)	$\mathcal{L}_{0.01}$ ($x10^{33}\mathrm{cm}^{-2}\mathrm{s}^{-1}$)
baseline 3 TeV	5.9	2.0
Guinea-Pig	5.93	2.1
Guinea-Pig++	5.93	2.1

- Both versions predict comparable ($O(10^{-4})$) values for e^+e^- luminosities
- The only discrepancy comes up in $\gamma\gamma$ interactions, which can be due to limited statistics

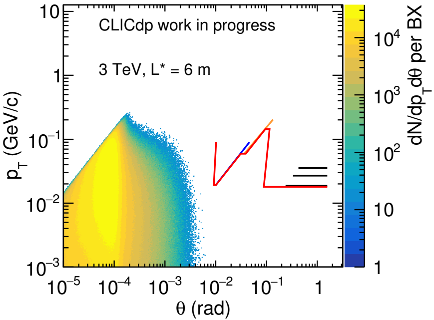
Background yields in 3 TeV CLIC with L* = 6 m

background type	Guinea-Pig	Guinea-Pig++	unit
beamstrahlung γ	2.04	2.06	γ per beam particle
incoherent e^+e^- pairs	3.1	3.1	$(\cdot 10^5)$ per bunch crossing
coherent pairs	3.2	3.2	$(\cdot 10^7)$ per bunch crossing
$\gamma\gamma \rightarrow$ hadrons events	2.9	3.0	events per bunch crossing

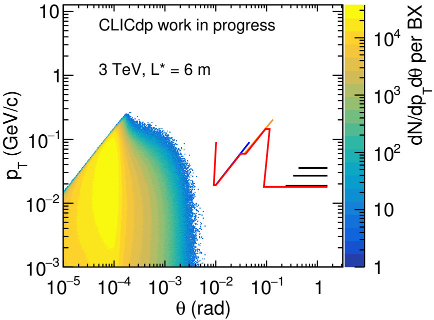
- Both versions predict comparable values for background yields
- The $\gamma\gamma \rightarrow$ hadrons discrepancy is linked with the luminosity spectrum difference; might be a statistics issue

Beamstrahlung photons distributions comparison

Guinea-Pig



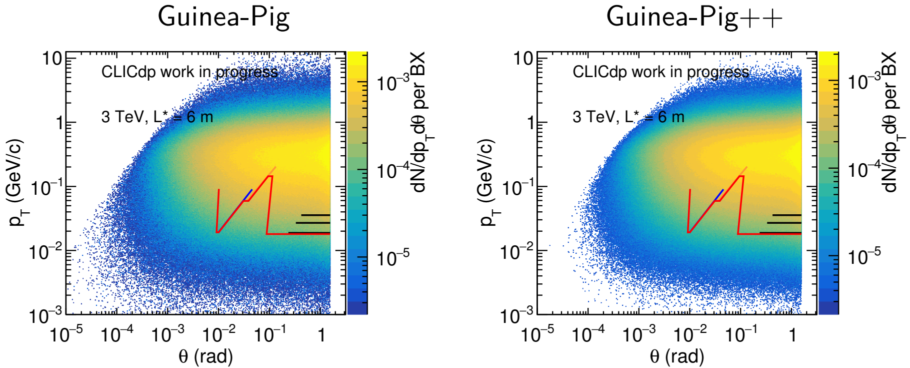
Guinea-Pig++



- The distributions of beamstrahlung photons are in good agreement between the two codes
- Beamstrahlung distribution is not a source of direct background, as required by the detector and delivery system designs

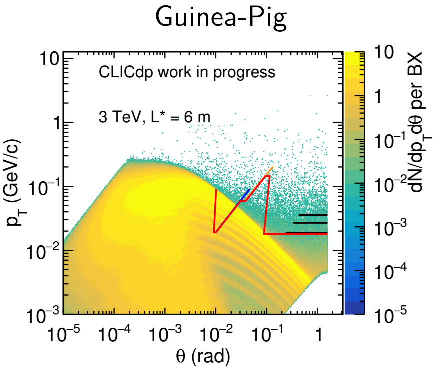
s

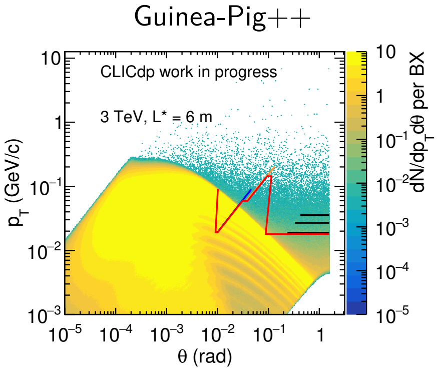
$\gamma\gamma \rightarrow$ hadrons distributions comparison



- $\gamma\gamma \rightarrow$ hadrons events are in agreement in both codes, no significant discrepancies are found
- This background gives rise to increased occupancies in the detector

Incoherent pairs distributions comparison

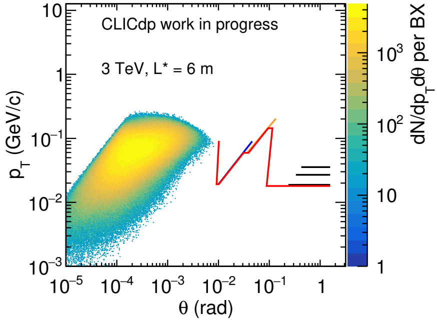




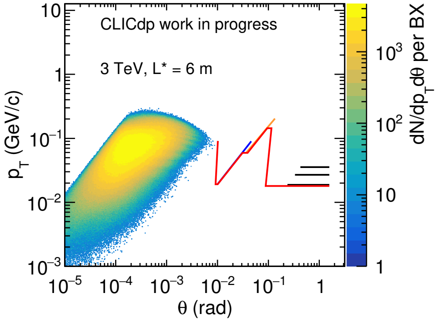
- Both distributions are comparable, the Guinea-Pig++ one has more statistics
- Incoherent pairs especially irradiate the forward region of the detector

Coherent pairs distributions comparison

Guinea-Pig



Guinea-Pig++



- Both distributions are comparable
- No direct hits are expected to be caused by coherent pairs
- The small differences in the distributions are attributable to the statistics effects

Guinea-Pig++ best practises

- It has been found that the C++ version sometimes produces highly deflected particles although it can be mitigated using a more strict approach to the input parameters and the beam quality, as advised by Barbara Dalena
- The grid size should be chosen to minimise the number of particles outside of it, depends on the beam quality and presence of tails
- Best longitudinal cut is in range of 3-3.35 σ_z, if it is too big some slices will have no charge
- Mesh granularity should be in range of 5-10 times the cut value in sigmas
- Two-beam simulations should be used for physics studies whenever possible, and leave one-beam for testing/debugging due to correlations between grid sizes and e.g. computed luminosity, which can be diluted this way, for more details see: CLIC Beam Physics Meeting 2 March 2017

Guinea-Pig developments since the last reports

- Truncation of numbers in output files led to rise of numerical errors and unwanted correlations
- The precision has been changed from 3 digits to 8 at the cost of large increase of storage space the output takes - can be solved by changing the output from ASCII to binary files
- Change in Guinea-Pig++'s output for beamstrahlung photons to the one compatible with Guinea-Pig - added the information about photons positions

Summary

- Software validation of Guineapig++ and Guinea-Pig has been done and the codes give results that are in agreement with each other, except for a slight difference in $\gamma\gamma \to$ hadrons events which has not been observed before and can be due to limited statistics available
- One needs to take into account I/O differences when running simulations and parsing through the outputs
- All current results regarding background yields, luminosity and the files produced by Guineapig++ are available at the revived Beam-Beam website: Beam-beam website

Outlook:

- Migrate Guinea-Pig and all related CLIC Beam Physics repositories from SVN to GitLab and make them available to run on the Grid using CVMFS installations

Thank you!

Backup