Gaussino - a Gaudi-based core simulation framework

Benedetto G. Siddi\(^{(1)}\), D. Müller\(^{(2)}\)

On behalf of the LHCb experiment

\(^{(1)}\)INFN - Sezione Ferrara
\(^{(2)}\)CERN

IEEE/NSS 2019
Manchester, UK
Motivation

- Starting point: general modernisation of LHCb software
- Main concept: separate core functionality for simulation
  - Experiment-independent framework: LHCb based algorithm built on top of that

Current Gauss building block

- ~15 years old
- Single-threaded framework
- Memory usage becoming an issue
- Simulation based on Geant4
- Increase in CPU doesn’t translate into an increase in memory
Main Features

- Keep the design choices of current Gauss, e.g.:
  - Integrated generation and simulation phase

- Include the following features:
  - Multi-threaded Gaudi
  - Can benefit from MT generators
  - HepMC3
  - Improve easy python configuration for jobs
  - Multi-threaded Geant4

New Framework built dependencies

Ref. [LHCB-TDR-017]
Gaussino

- A complete simulation framework following the basic Gauss architecture

Generator phase with Pythia8 interface implemented
Code migrated to HepMC 3

Simulation phase under redesign with identification of elements.
Aim to keep it ‘simulation engine’ independent

Ref. [link]

Event Loop controlled by Gaudi
Detector geometry from same source as other applications
**Execution structure**

Gaudi Functional

- Every Algorithm as task
- Constant execution
  - Random Engine created per call
- Fixed input/output, e.g.

```cpp
/*output data*/ operator()(const /*input data*/) const
```

Ref. [Task based scheduler]
Multi-Threading

- Thread safety of generators
  - Production and decay tools
- Example in Gaussino: Pythia 8
  - Shared, locked
- Thread local

---

**Generation Phase Only!**

---

**LHCb simulation**

- P8: 2.73 MB/thread
- P8MT: 11.18 MB/thread

**Throughput [a.u.]**

- MB P8
- MB P8MT

---

**Maximal memory [GB]**

- LHCb simulation

---

**B. Siddi**

INFN Ferrara

Gaussino - a Gaudi-based core simulation framework
Modular Simulation

- Take inspiration from the generation: modular!
- A simulation service managing different backends
- Flexible configuration, e.g. different setting for:
  - Pile-up - Spillover - Main Event
  - Signal - other particles
  - Fast Simulation
Simulation Interface

• Gaussino - factories
  ○ Easy python configuration of G4 settings
• Dynamic assignment possible
  ○ Entire Gaudi event
  ○ Split Gaudi Event into multiple G4 workloads
  ○ Plan to investigate new track based parallelism (in G4)
Truth Matching

- Geant4 itself does not have any truth information

In current Gauss
- Writing HepMC structure keeping the history using G4 track actions
- Relying on global state for communications
- After simulation: converted to MCParticles (object storing the truth in LHCb Event Model)

For Future
- Redesign for multi-threading
- Simpler: directly write MCParticles

Ref. [HepMC]
Detector Geometry

- Move to **DD4Hep** for the detector wide translation
- Requires a bit redesign in simulation

**User must provide:**
- A Gaudi tool that provides the world pointer
- Logical volume to sensitive detector mapping (configurable via python)
- Basically python configurable UserDetectorConstruction
- DD4hep prototype being tested

**Benefits**
- Converter for loaded DD4Hep geometry to Geant4 already exists
  - Simplifies the GaussGeo conversion service
- Perform all modifications of geometry in DD4Hep

Ref. [DD4Hep]
Future Gauss prototype

- Ported additional components
  - EvtGen
  - LHCb geometry description (DD4Hep in progress)
- Added to automatic LHCbPR testing (Ref. [Link])
Gauss + Gaussino
Migrated generation tools

- EvtGen (single, locking instance)
- Handles all decays
- Logic for signal generation
- LHCb specific generator cuts
  - Not affecting the performances

**Generation Only** \( D^{*+} \rightarrow D^{0} \pi^+ \) with \( D^{0} \rightarrow K^- \pi^+ \) events
Gauss + Gaussino + Geant4
Validation studies

- Performed in LHCbPR (the LHCb performance and regression monitoring tool)

Energy deposit (Generation + Simulation) \( D^{*+} \rightarrow D^0 \pi^+ \) with \( D^0 \rightarrow K^- \pi^+ \) events
Performances

- Tested with locking/thread-local generation phase
- Overall good scaling in throughput
- Very good scaling in memory usage

Tested with typical tagged D0 decays with 2016 conditions

Generation + Simulation $D^{*+} \rightarrow D^0 \pi^+ \text{ with } D^0 \rightarrow K^- \pi^+$ events
Conclusions

- Gaussino, an experiment-independent core framework developed in LHCb in very advanced state.
- Modular generation phase with Pythia8 example
- Gauss prototype built on top
- Interaction with Geant4 MT
- Encouraging first results, i.e.:
  - Performances not affected by LHCb specific tools
  - Almost perfect matching between old and new simulation
  - Higher throughput