Geant4 vs ONNXRuntime simulation times in ddsim

A first look

Software and computing setup

- /cvmfs/ilc.desy.de/key4hep/setup.sh
  - 2023-03-15 version of the stack
- https://gitlab.desy.de/ilcsoft/ddfastshowerml
  - `cmake .. -DCMAKE_INSTALL_PREFIX=../install -DCMAKE_CXX_STANDARD=17 -GNinja`
  - `ninja install`
  - Add <workdir>/install/lib64 to LD_LIBRARY_PATH
  - **Need to add ONNXRuntime to LD_LIBRARY_PATH** (for now, to be checked):
  - export LD_LIBRARY_PATH=/cvmfs/ilc.desy.de/key4hep/spackages/py-onnx-runtime/1.7.2/x86_64-centos7-gcc11.2.0-opt/6l75cuhoooj4w63mlar227g6pwrkqjxm/lib64/:$LD_LIBRARY_PATH
- Running locally on my laptop inside a CentOS7 container (via singularity)
  - 16 GB RAM, i7-9750H @ 2.60GHz (6x2 cores)
- Using EDM4hep output for ddsim (LCIO broken in DD4hep version)

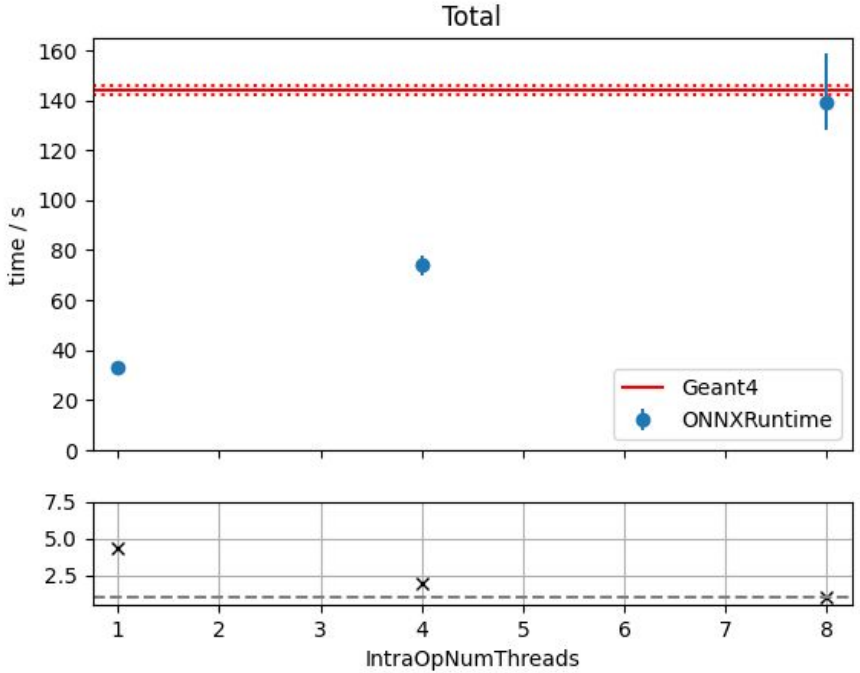
Physics and ML model setup

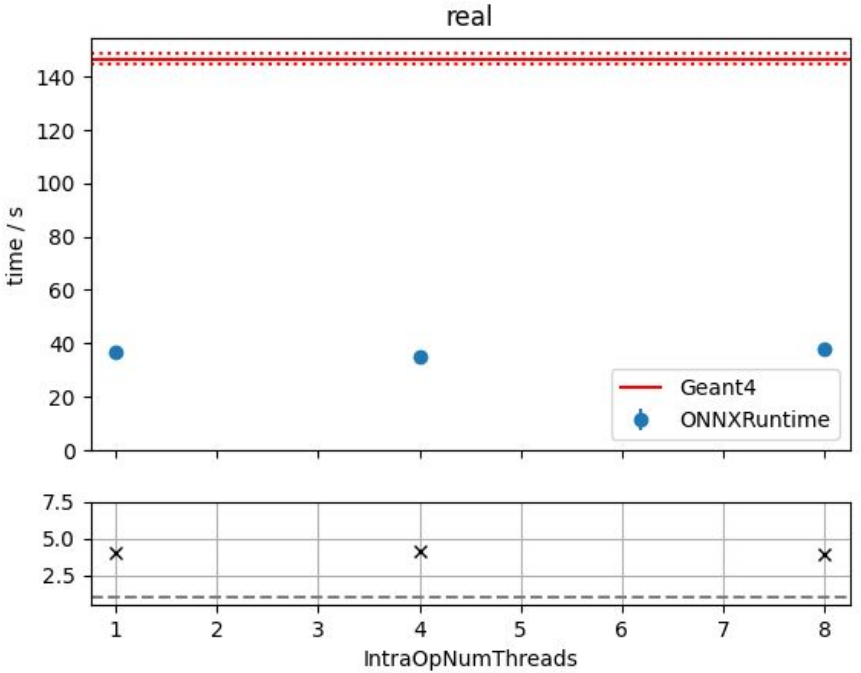
- Shooting single photos @20 GeV with random direction
- ILD_l5_o1_v02 geometry
- 100 events
- Using Franziskas GAN model
- Frank has implemented the necessary functionality for
  - Rotating and translating the regular grid GAN output to the correct place in the detector
  - Conversion to “SpacePoints” + hand off to Geant4 for placing them in geometry
- For now only using ONNXRuntime (no Torch in Key4hep yet)
- Varying the number of CPU threads the ONNXRuntime can use (*IntraOpNumThreads*)

Running the benchmarks

- Using the time command and ddsim output to collect run times
- Run each benchmark 3 times, use min/mean/max time in plots
- **Very preliminary benchmarks!**
  - No CPU (and/or NUMA) pinning
  - Other SW running on the same machine
- Should be OK for some first insights

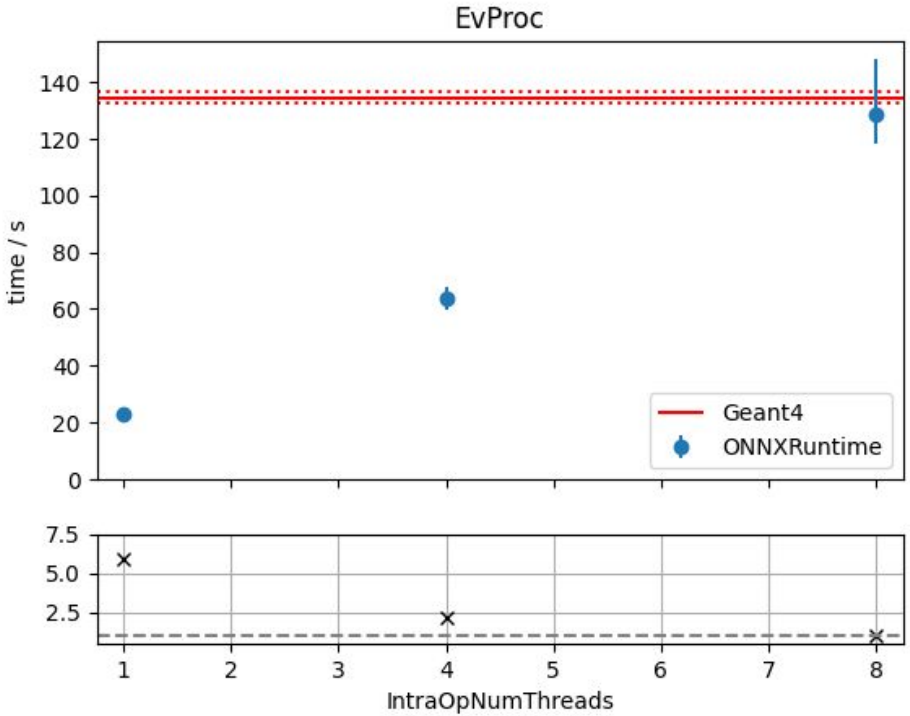
Total and real run time

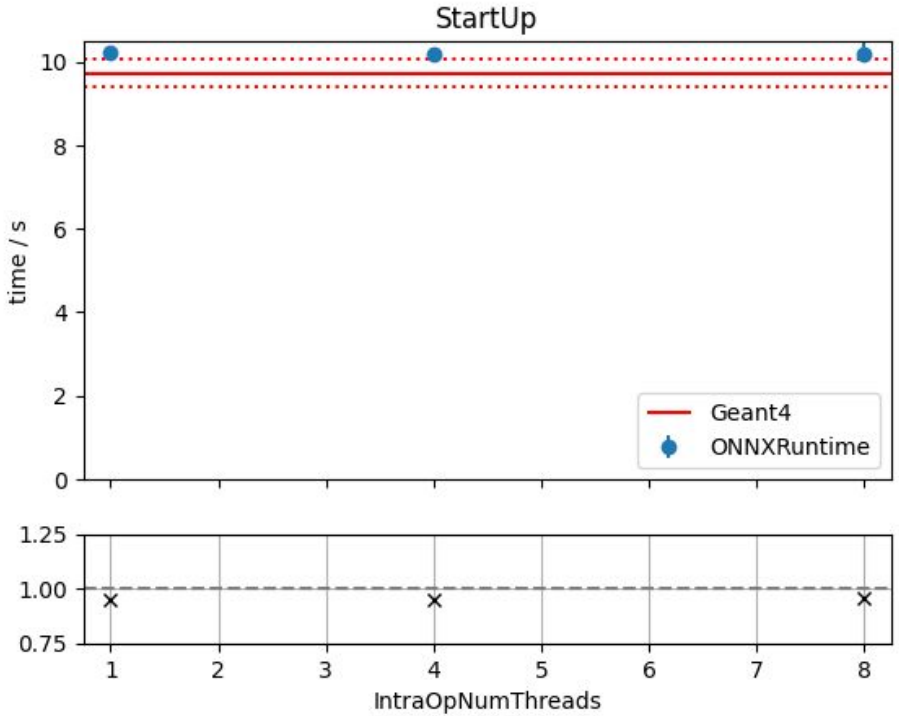




Total = As reported by ddsim (aka total CPU time, corresponding to *user*)
real = Real time it took to execute the command (aka observable by the user)

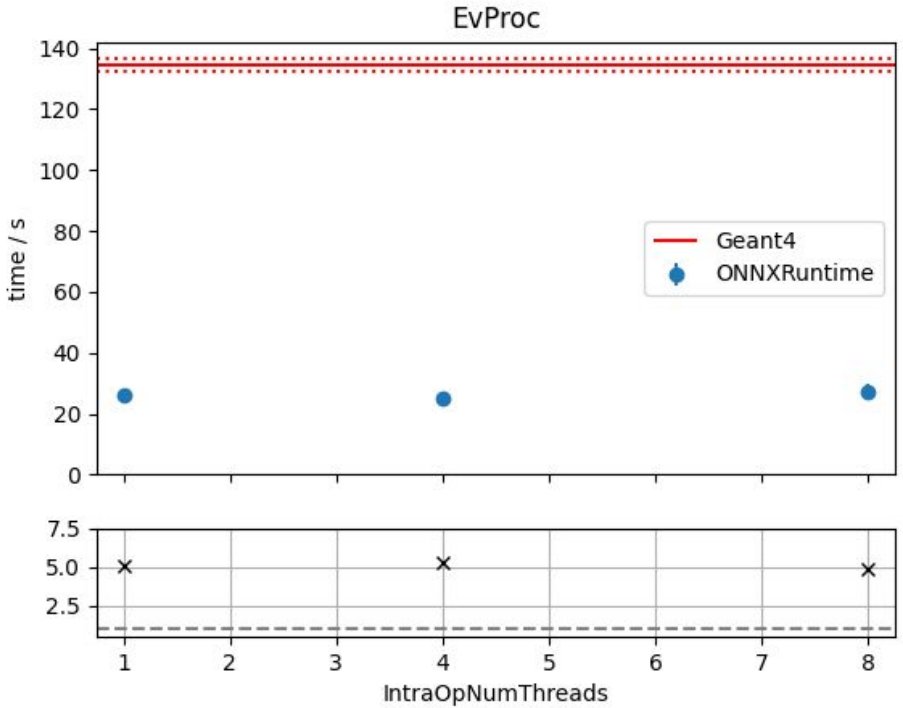
StartUp and per event times





Event Processing times as reported by ddsim (i.e. CPU time)

“Real” per event times



- DDSim records CPU time
- Using real - StartUp time to get to “observable” per event times

Summary / Conclusions

- Can run Franziskas GAN via ddsim for ECAL showers in ILD
- Approx. 4X speedup (total time) wrt. Geant4
- Approx. 5X speedup (per event time) wrt. Geant4
- ONNXRuntime seems to not scale at all with number of threads
- Very preliminary results! Still need to check whether there are easy optimizations somewhere