

Update on ImPACT with ctapipe

Thomas Gasparetto

(University of Trieste & INFN,
Université Grenoble Alpes & LAPP)

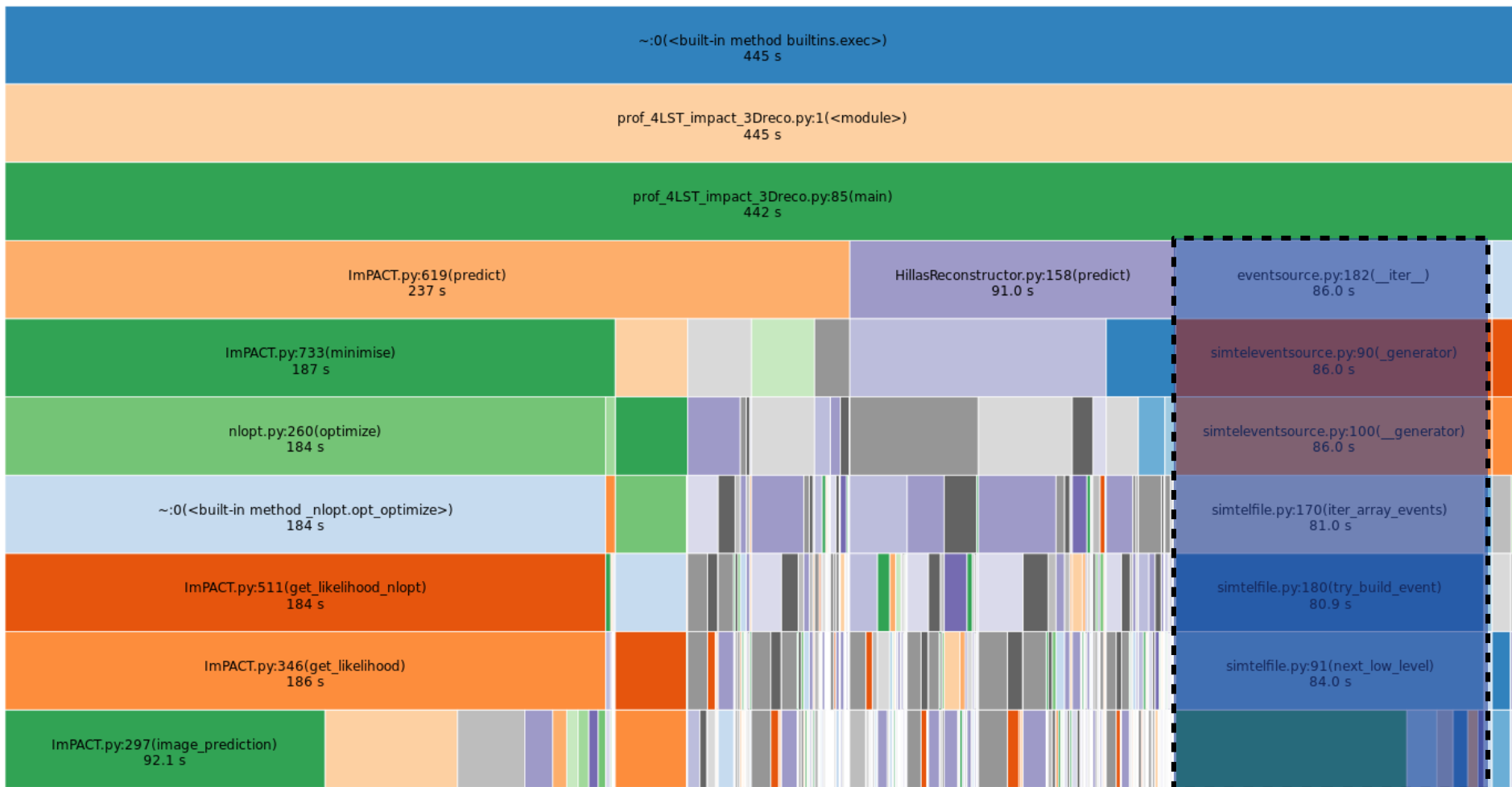
To-do list

- I had to move from the 2D reconstruction method to the 3D method present in ctapipe. A bit of cleaning to the pipeline...
- 3D method as seed to ImPACT since it gives better performances with respect to the 2D.
- Clean-up and speed-up of ImPACT (PR #946):
 - Goal: speed-up ImPACT.
 - How: profile & check results at each “optimization step”

Final goal:

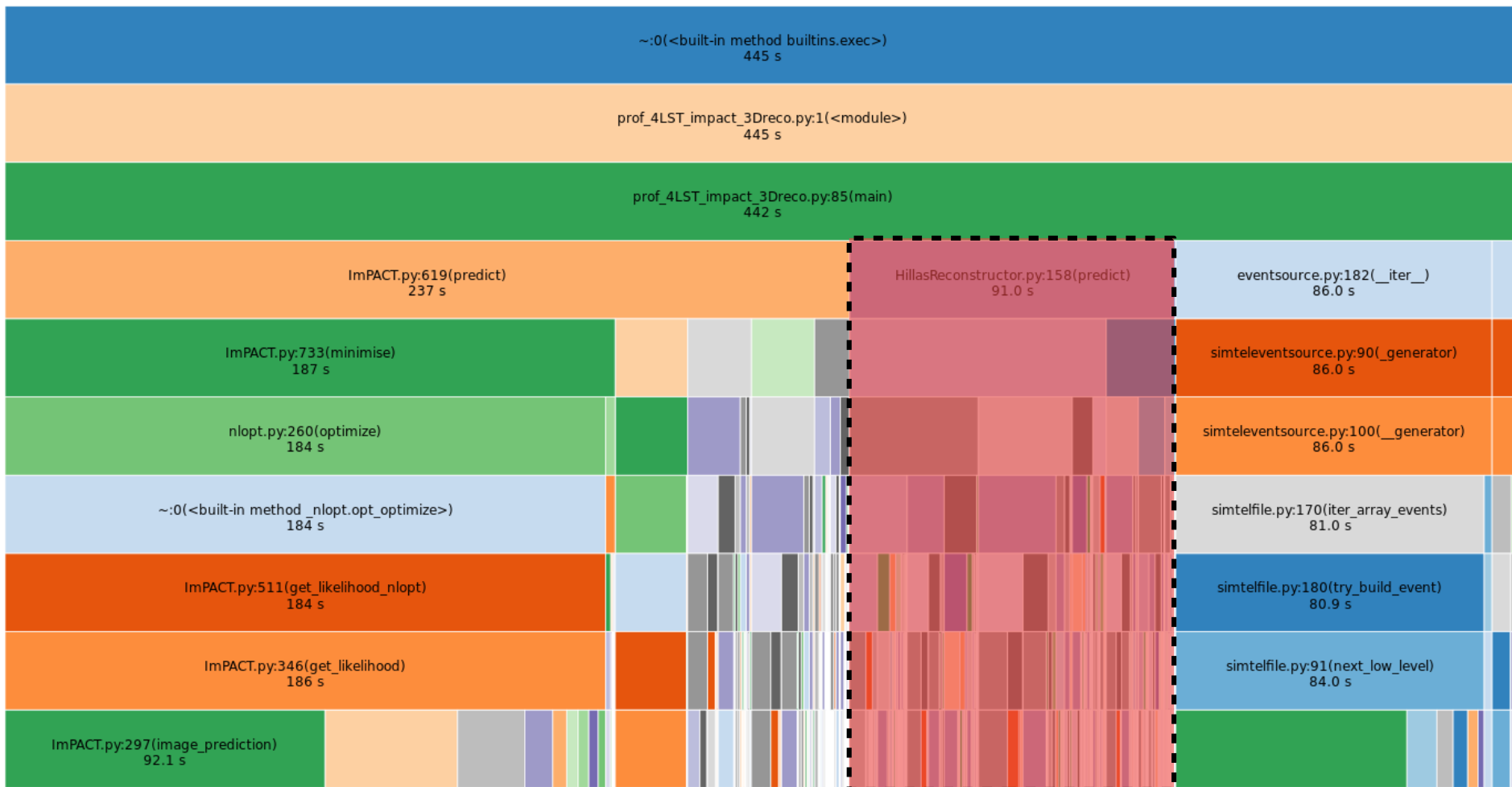
- Move ImPACT to GPU thanks to pytorch
- Not an easy task but this would offer the speed-up together with good physical performances

The profiling



I/O

The profiling



3D reco

The profiling

~:0(<built-in method builtins.exec>)
445 s

prof_4LST_impact_3Dreco.py:1(<module>)
445 s

prof_4LST_impact_3Dreco.py:85(main)
442 s

ImPACT.py:619(predict)
237 s

HillasReconstructor.py:158(predict)
91.0 s

eventsource.py:182(__iter__)
86.0 s

ImPACT.py:733(minimise)
187 s

simteventsources.py:90(_generator)
86.0 s

nlopt.py:260(optimize)
184 s

simteventsources.py:100(_generator)
86.0 s

~:0(<built-in method nlopt.opt.optimize>)
184 s

simtelfile.py:170(iter_array_events)
81.0 s

ImPACT.py:511(get_likelihood_nlopt)
184 s

simtelfile.py:180(try_build_event)
80.9 s

ImPACT.py:346(get_likelihood)
186 s

simtelfile.py:91(next_low_level)
84.0 s

ImPACT.py:297(image_prediction)
92.1 s

ImPACT.predict

- create physics benchmark

ImPACT.py:619(predict)
237 s

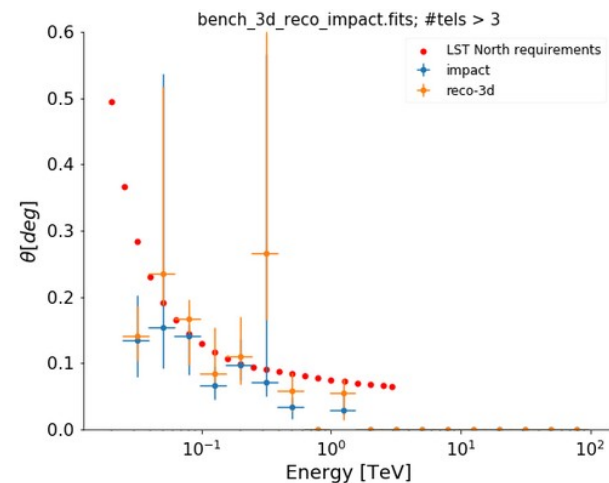
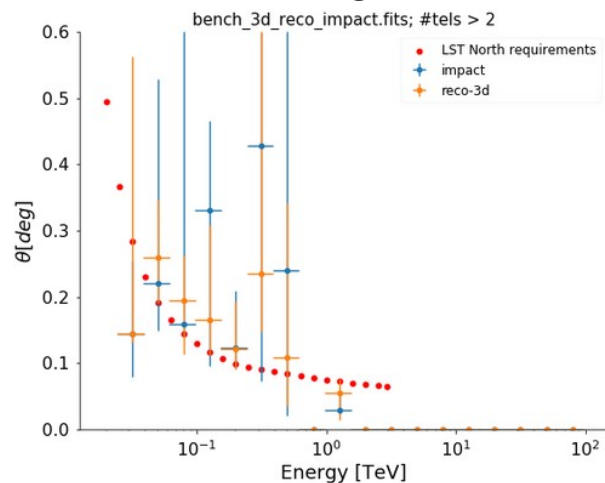
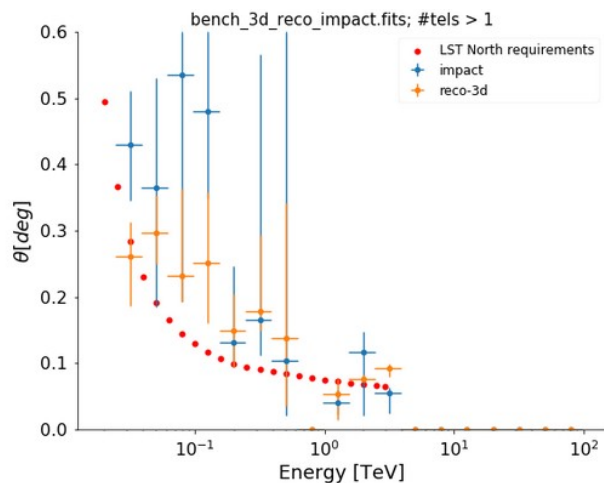
ImPACT.py:733(minimise)
187 s

nlopt.py:260(optimize)
184 s

~:0(<built-in method _nlopt.opt_optimize>)
184 s

ImPACT.py:511(get_likelihood_nlopt)
184 s

4 LSTs, Paranal, gamma, z20

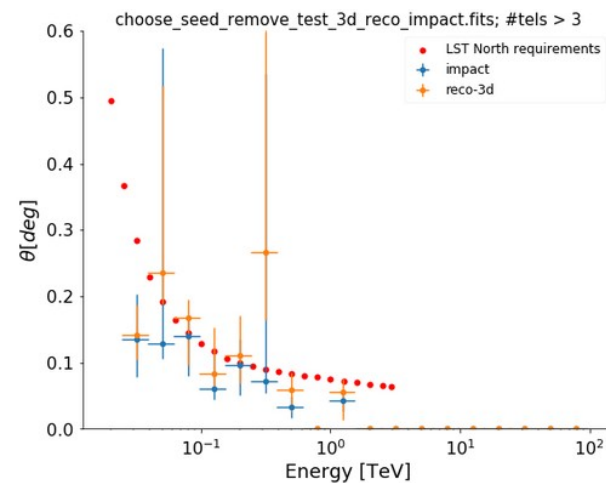
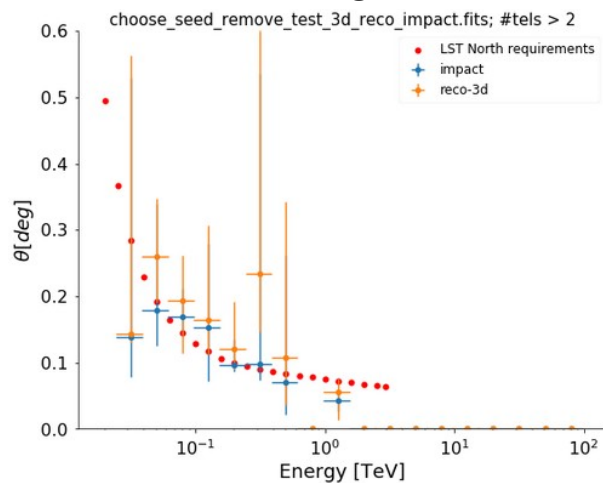
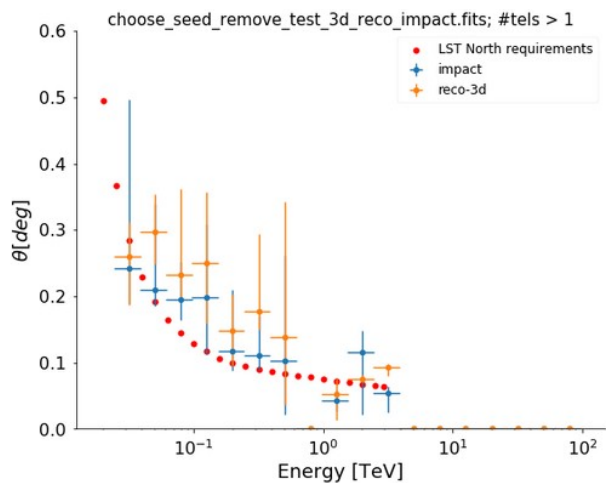


ImPACT optimization - 1

- Remove choose_seed function which re-choose the seed among many seeds (if #telescopes < 4)

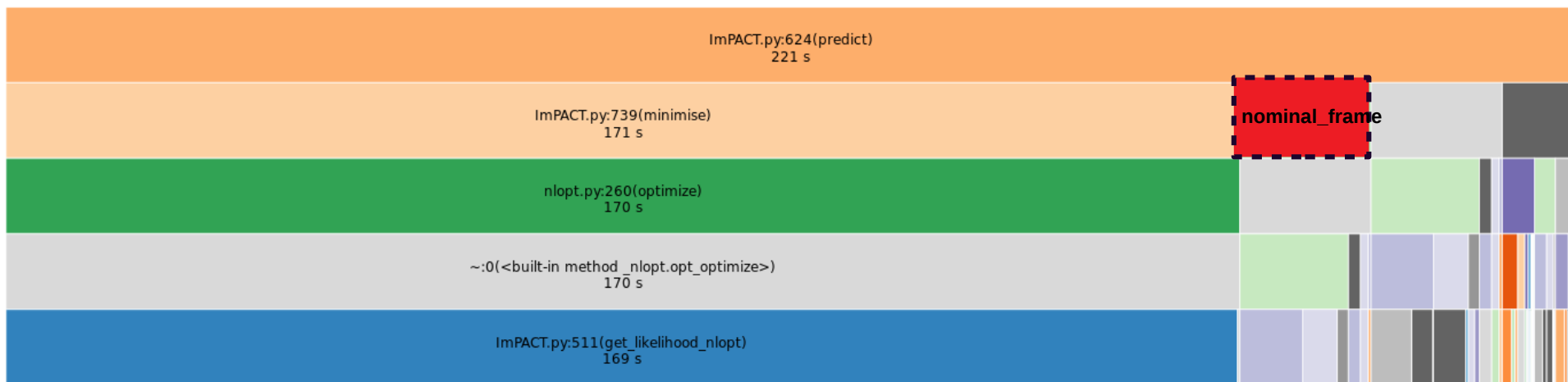


4 LSTs, Paranal, gamma, z20

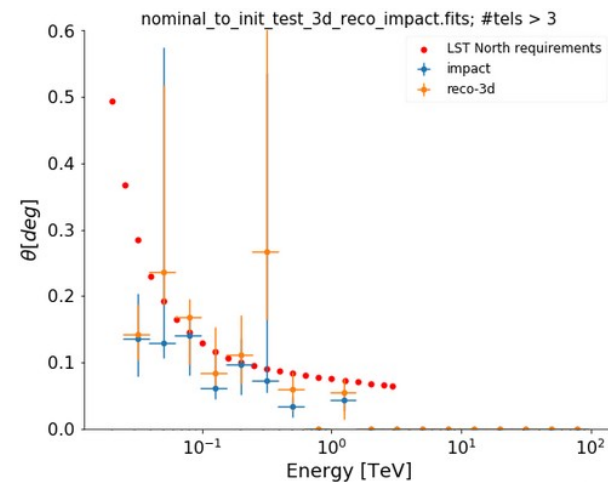
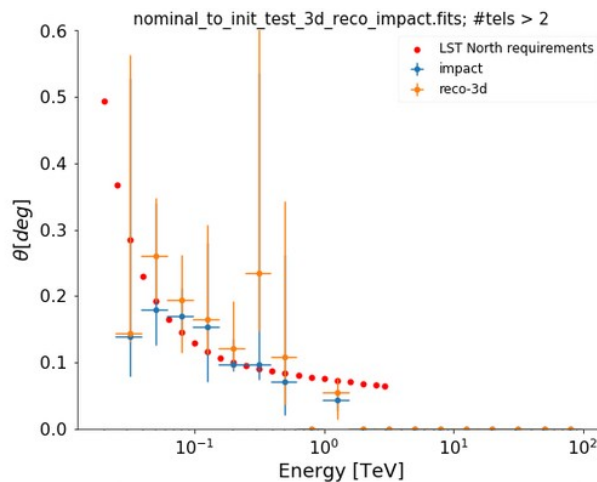
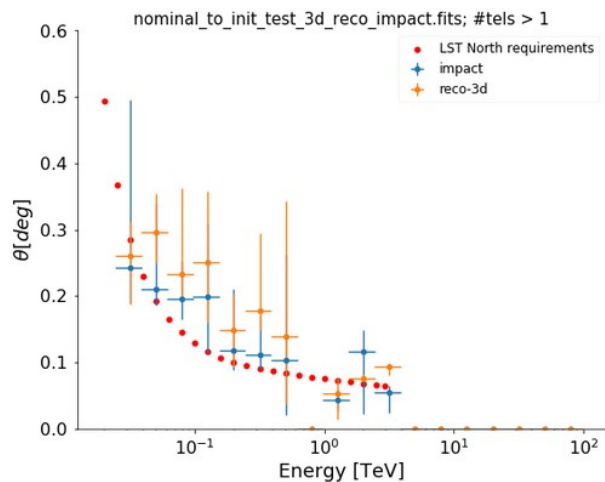


ImPACT optimization - 2

- Remove nominal frame creation at each function call..it's the same for all the events

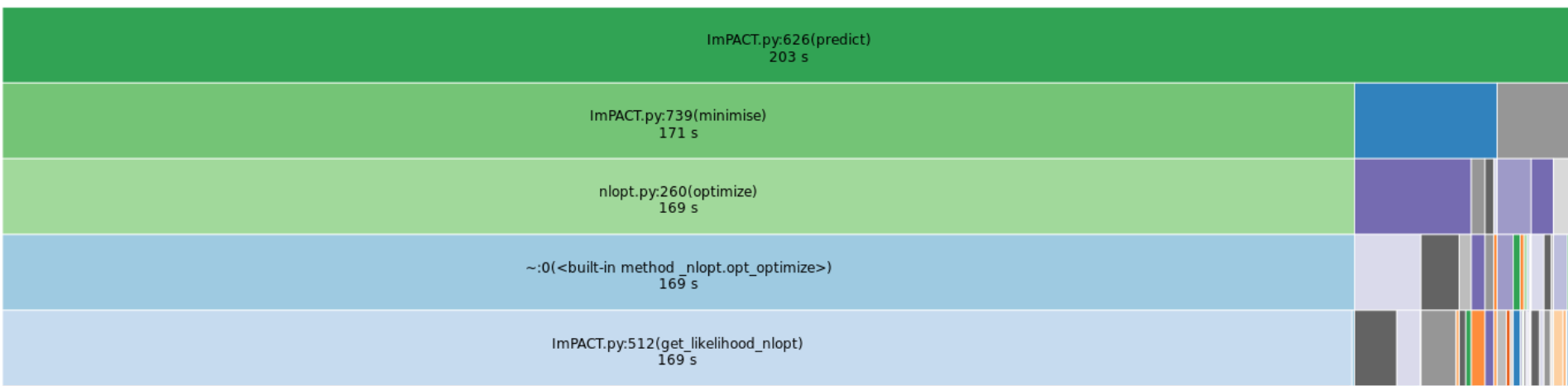


4 LSTs, Paranal, gamma, z20

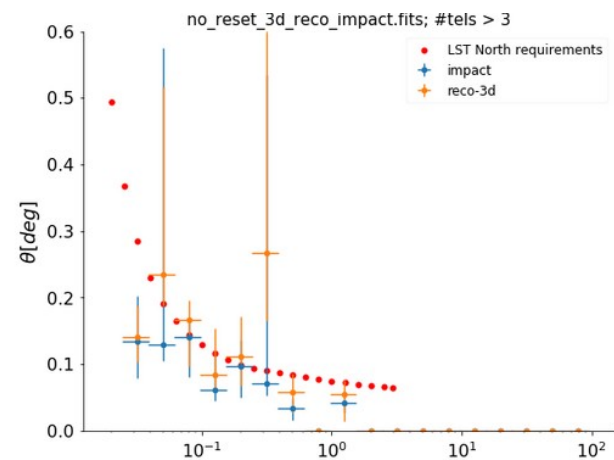
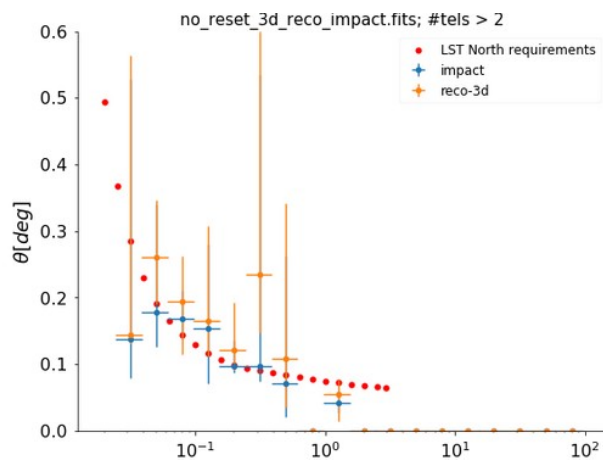
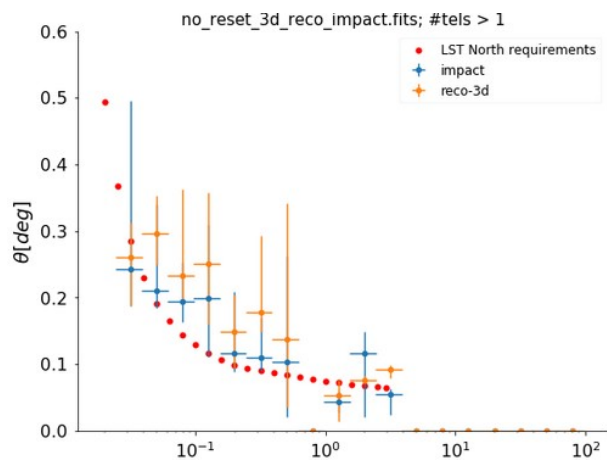


ImPACT optimization - 3

- the overall ImPACT part is 15% faster with much better results...it's a double win!



4 LSTs, Paranal, gamma, z20



Next steps...

- Final goal:
 - Better physics performances
 - Low overhead with respect to the standard analysis
- A bit more of work on the ctapipe optimization
- Many ideas on how to port it on GPU