



PR and release status: BSM processes (HEFT), Floating Point Exceptions etc.

Andrea Valassi (CERN)

Madgraph on GPU development meeting, 30th April 2024

<https://indico.cern.ch/event/1355151>

(these are the slides I had prepared for the April 15 meeting, which was cancelled)

(previous update was on April 04)

Status update for my PRs – BSM

- Update on my PRs and issues (since April 04 meeting)
 - **MERGED: PR #824 (SUSY)** – reviewed OM
 - Still pending *issue #825 (susy_gg_tt madevent tests – xsec mismatch Fortran vs cudacpp)*
 - Still pending *issue #826 (susy_gg_t1t1 madevent tests – no xsec in cudacpp madevent)*
 - **MERGED: PR #632 (SMEFT and HEFT)** – reviewed OM
 - Fixed various issues (614, 616, 633) blocking gg>tt~tt~ for CMS and Zenny’s reweighting
 - Pending minor issue *#827 (determine number of BSM parameters when generating CPPProcess.cc)*
 - *New issue #828 (heft_gg_bb has no FFS2_0 calls, unlike heft_gg_h_bb) now fixed in later PR #832*
 - SM gg>bb~ has 3 diagrams, HEFT gg>h>bb~ adds 1 diagram, but HEFT gg>bb~ has only 3 instead of 4?
 - Solution is to add “MIW<=1” to keep the heavily suppressed gg>h>bb~ diagram (thanks OM!)
 - **New: PR #832 (HEFT and FPEs)** – complete and ready to merge, awaiting 2nd review by OM
 - After fixing #828 above, I added heft_gg_bb and removed heft_gg_h from the repo and tests
 - Advantage: was finally able to test madevent for HEFT (different events with different momenta)
 - *New issue #833 (LHE file mismatch Fortran vs cudacpp, for FPTYPE=f only)*
 - To be investigated: there might be nothing to be fixed (maybe due to loss of precision in Feynman for float?)
 - First review done with OM, implemented various changes in handling FPEs (next slide)
- Summary of BSM status (when PR 832 is eventually merged)
 - SUSY models ~OK (except for issues 825 and 826 above)
 - HEFT models ~OK (except for issue 832 above)
 - SMEFT models look OK

Floating Point Exceptions (FPEs) revisited

- **Motivation: UNDERFLOW #831 in heft_gg_bb tests in new PR #832 (FPTYPE=f,m)**
 - The HEFT gg>h>bb~ diagram is suppressed with respect to the three SM gg>bb~ diagrams
 - Some amplitudes sum up to below E-19: their square is below E-38, which causes underflow
 - My first attempt: manually flush to zero amplitudes < E-19... now removed after discussing with OM
- **Reminder: had seen INVALID sqrt(-1), DIVBYZERO and OVERFLOW in pp>ttW**
 - Now fixed, after *a lot* of work to improve SIMD ixxx/oxxx (#701, #733, #736, #738...)
 - Printed out when cudacpp linked with Fortran madevent (“IEEE_DIVBYZERO is signalling”)
 - To debug in cudacpp: had enabled SIGFPE with feenableexcept (if an env variable was set)
 - Had enabled all four INVALID, DIVBYZERO, OVERFLOW and also UNDERFLOW
- **New understanding (what I had not understood before)**
 - One: the Fortran messages (“IEEE_DIVBYZERO is signalling”) are warnings, not errors
 - Two: for UNDERFLOW itself, results are “denormal” (inexact), but better than zero...
- **New strategy implemented in PR #832**
 - **Do enable SIGFPE for INVALID, DIVBYZERO, OVERFLOW: they will cause a crash**
 - These must be considered as errors – and they have normally all have been fixed?!
 - **Do not enable SIGFPE for UNDERFLOW: this is acceptable, should not cause a crash**
 - However, keep track of UNDERFLOWS and print a warning at the end of cudacpp programs
 - In addition to the “IEEE_UNDERFLOW is signalling” printout from Fortran if there is one

Status update for my PRs – others

- Update on my PRs and issues (since April 04 meeting)
 - **No update: PR #798 (Makefile targets)** – was and still is ready to merge, awaiting review
 - This completely separates C++ and CUDA or HIP targets (extends/completes Jorgen's earlier PR)
 - Merged again with latest upstream/master including SUSY/SMEFT PRs
- Next priority for me: git repos and transfer scripts
 - See also the discussion in issue [#661](#) and in Olivier's issue [#815](#)
 - There was some discussion at the general mg5amcnlo meeting about git subrepos

A few other comments

- Olivier's PR [#835](#) (default `FPTYPE=m` in `runcard`) – AV review, looks ok for me
 - However, I would suggest adding also `HELINL` and `HRDCOD` to `runcard` (issue [#700](#))
 - And then make the syntax more consistent (e.g. `cudacpp_fptype`, `cudacpp_helinl`, etc)
 - Note also: pending PR [#798](#) removes 'avx' ('backend' is enough to track avx in `runcard`)
- WIP on warps and channel ID arrays (issue [#765](#))
 - New: Stefan's PR [#830](#) from the [new interface wrap](#) branch
 - Based on Olivier's `mg5amcnlo` [gpucpp_wrap](#) branch
 - Question: do (can) we have tests specifically for the new functionality?
- Nathan's Intel GPU support in `cudacpp` ([#805](#)): maybe after the release?
- Process-specific issues on AMD GPUs: segfault in `gq_ttq` ([#806](#))
 - I suggest we release without waiting for this and we fix it later
- Various other comments from May 2023 issue [#671](#)
 - In general: test, test, test (especially the full workflow, and parameter/run cards etc)
- Check what else we still have against `madgraph4gpu` (e.g. Lugano) before closing it