



PR and release status BSM processes (SUSY, EFT) etc.

Andrea Valassi (CERN)

Madgraph on GPU development meeting, 19th March 2024

<https://indico.cern.ch/event/1355148>

Progress with BSM PRs – summary (after PR [#824](#))

- **SUSY models: ~OK** (SM processes with SUSY parameters, SUSY processes)
 - Fixed many issues, mainly in handling of α_S -indep real parameters and complex couplings
 - The issues were generally in the computation of derived α_S -dep parameters and couplings
 - Code generation is OK (with no check that model name includes ‘sm’, as suggested by OM)
 - For both HRDCOD=0,1: builds and basic cudacpp tests (check.exe, runTest.exe) are OK
 - *Not OK: susy_gg_tt madevent tests (xsec mismatch Fortran vs cudacpp, issue [#825](#))*
 - *Not OK: susy_gg_t1t1 madevent tests (no xsec in cudacpp madevent, issue [#826](#))*
- **HEFT models: OK?** (only tested heft_gg_h so far)
 - Codegen OK, builds and basic tests OK (new: also for HRDCOD=0, which previously failed)
 - *Not tested: HEFT madevent tests (gg_h has no degrees of freedom in the phase space)*
 - *Can you suggest a better process? Maybe HEFT gg to bb with non-zero b mass?*
- **SMEFT models: NOT OK** (testing smeft_gg_tttt in PR [#632](#))
 - Codegen OK
 - *Not OK: neither HRDCOD=0 (issue [#616](#)) nor HRDCOD=1 (issue [#614](#)) builds*

(Reminder: interest at least in CMS for SMEFT LO and in ATLAS for SUSY LO)

Progress with BSM PRs – newly merged

- PR [#822](#) (AV) – **MERGED** (approved OM)
 - Fix bug in GPUFOHelasCallWriter `format_coupling` (fix [#821](#))
 - Different parameters were assigned the same index in `params2order`, e.g. affecting SUSY
- PR [#625](#) (AV, fixes for SUSY) – **MERGED** (approved OM)
 - Fixed SUSY `gg_tt` builds and `check/runTest` in C++ and CUDA (both `HRDCOD=0` and `=1`)
 - This PR fixes only a SM process (gg to top pair) modified with SUSY parameters
 - NB: not tested (and not fixed) in this PR: Fortran vs cuda/cpp result comparison
 - Main fixes are in *BSM double α_S -indep parameters* used for computing α_S -dep couplings
 - `HRDCOD=1`: add `constexpr` implementation of `sin/cos/tan` based on Taylor series (fix [#627](#))
 - `HRDCOD=0`: fixed parameter visibility (e.g. `mdl_I51x11`) and copied them to GPU constant memory
 - Having `HRDCOD=1` is useful also to compare results! e.g. zero MEs for `HRDCOD=0` (fix [#818](#))
 - NB: not yet fixed by this PR (fixed in later PR): handling of BSM complex α_S -indep couplings
 - Completed the backport to CODEGEN of these fixes
 - NB: not yet fixed by this PR (fixed in later PR): proper SUSY processes like gg to stop pair
 - NB: not yet fixed by this PR (largely still open): many issues still pending for EFT processes...

Progress with BSM PRs – ready to merge

- PR [#824](#) (AV, fixes for SUSY/HEFT) – **READY TO MERGE** (review OM in progress)
 - Fixed SUSY `gg_t1t1` builds and `check/runTest` in C++ and CUDA (both `HRDCOD=0` and `=1`)
 - This PR fixes many true SUSY processes: `gg` to stop pair, `gg` to gluino pair, `gg` to squark pair
 - Main new fixes are in *BSM complex α_S -indep couplings* used for computing α_S -dep couplings
 - These are tested in `susy_gg_t1t1` (added to the repo),
 - *BSM double α_S -indep parameters* were tested in `susy_gg_tt` (previously added to the repo)
 - Completed the backport to CODEGEN of these fixes
 - Also fixes HEFT `gg_h` builds and tests in C++ and CUDA for `HRDCOD=0` and `=1`)
 - NB: not yet fixed and still pending: many issues for SMEFT processes
 - After review by Olivier (thanks!): improved these fixes to never rely on model name ‘sm’
 - Exactly the same code generation is done for SM and BSM now
 - Still to be improved: get rid of `MGONGPUCPP_NBSMINDEPPARAM_GT_0` [#827](#)
 - The number of “additional BSM parameters” is known in `Parameters.h` but not when generating `CPPProcess.cc`
 - Added at the end of this PR: `madevent` tests (e.g. Fortran vs `cuda/cpp` result comparison)
 - *Identified issue [#825](#): SUSY `gg_tt` cross section differs in Fortran and `cuda/cpp` – not fixed yet*
 - *Identified issue [#826](#): SUSY `gg_t1t1` has no cross section in `cuda/cpp` `madevent` – not fixed yet*

Other PRs – almost ready or WIP

- PR [#819](#) (NN, latest SYCL branch) – **READY TO MERGE?**
 - Latest changes to `epochX/sycl` (this does not affect the `cuda` directory)
 - (Sorry Nathan maybe I should have already merged this?)
- PR [#798](#) (AV, based on Jorgen's [#775](#)) – **~ALMOST READY (must fix new conflicts)**
 - Separate build targets for `CUDA` and `C++` (and must now add `HIP`)
 - This was complete and ready to merge before recent merges
 - Now there are a few ~easy conflicts to fix (`HIP`, `HIPRAND`, `gXXX.cu` all changed makefiles)
 - **One infrastructure issue: no AMD GPUs (LUMI access for CERN expired, being renewed)**
- Issue [#765](#) (SR's `new_interface_wrap` branch, no PR yet) – **WIP?**
 - From scalar channel ID to array of channel IDs
 - Eventually need also Olivier's `mg5amcnlo gpucpp_wrap` (not yet in `gpucpp`): complete?

Also missing before the release (non-exhaustive list?)

- **Update the separate plugin repo** (issue [#661](#)) or recreate it with the full history
 - [mg5amcnlo_cudacpp](#) exists with the full history but is stuck to Aug 30
 - As discussed two weeks ago: I will prepare scripts to copy commits to/from madgraph4gpu
- **Try to fix SUSY and EFT** before the release?
 - See extensive description of BSM status in the first three slides
- Wait for Nathan's **Intel GPU support?** ([#805](#))
 - Would also need the full manual tests on Intel GPUs (or even better an Intel GPU CI)
- Process-specific issues on **AMD GPUs: segfault in gq_ttq** ([#806](#))
 - I suggest we release without waiting for this and we fix it later
- See also the May 2023 summary (issue [#671](#)): not up to date but still relevant
 - The issues that are still open remain desirable, though not strictly necessary?
 - Many issues mentioned there have been fixed/completed
 - Only a few new issues have appeared (e.g. channel id array)
- Am I missing some very big thing not mentioned above?