Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2019-91-RC4, 2019

Geoscientific Model Development Discussions

Interactive comment on "Replicability of the EC-Earth3 Earth System Model under a change in computing environment" *by* François Massonnet et al.

Anonymous Referee #4

Received and published: 1 August 2019

General comments

This paper focuses on the EC-Earth model and considers how to demonstrate it is replicable across different HPC platforms. It adopts a statistical approach and demonstrates the use of metrics with a 'convenient' example of a potential bug in their user code.

The paper does not go into great detail on the causes of differences in the model results on different platforms. In practise, this investigation can be a long and tedious process to isolate the cause either in the user code or sometimes in vendor supplied code.

The technique may be applicable to other models, though it is difficult to tell, given the

complexity of the coupled EC-Earth ESM, what changes would be needed to the authors' methods to achieve reliable results. This paper presents an interesting example on an important issue and publication is recommended following minor revisions.

Specific comments

Page 3, Line 15 (and section 2.1) Sentence beginning 'On the other hand, . . . representation of numbers / operations can ..'. I find this sentence vague. The IEEE standard followed by vendors specifies how numbers are represented and rounded during arithmetic operations, unless a compiler is allowed to go beyond the standard, operations should not be expected to differ. Likewise, why would a different implementation of the MPI library (MPICH .v. OpenMPI) be expected to produce different results if MPI is simply used to move data between tasks? This paragraph appears to loosely suggest differences can arise simply from making these changes. There is no mention of the standards in place that compilers are expected to follow. Compiler options that violate standards are a user choice and often used, as the authors state, to achieve better performance.

Page 3, Lines 16 & 29. Issues in replicability/reproducibility can also arise from the operating system libraries in use, separate from the compiler. For example, optimized vendor supplied versions of the BLAS/LAPACK libraries, often used in ESMs, can give rise to differences compared to other implementations of these libraries. I suggest the authors reword to say 'compiler environment' rather than simply 'compiler' or 'compiler setup' wherever used.

Page 4, lines 2-3. Again, this is rather vague. The user/developer has a great deal of control over what the compiler is allowed to do in terms of optimizing arithmetic operations. To say '(or simply, the translation to assembly code...' is not correct. It is the code reorganisation performed by the compiler optimizations, then translated to assembler, which can be incorrect, either because of user code errors and/or inappropriate compiler options.

Section 3.1.2. The IFS model also supports OpenMP parallelization, can the authors clarify if OpenMP was used?

Section 3.1.4. Is the version of H-TESSEL used part of the IFS CY36, or a different version? Has it been modified from the version supplied with IFS?

Section 3.2. Why 20 years? Does this not depend on the choice of parameters studied?

Section 3.2.2. The IFS model normally outputs GRIB format files, which are a lossy compressed format. Can the authors clarify if they are using output at the precision of the model's arithmetic or some reduced precision format? This is important if looking for small differences and their results?

Page 10, table 2. Several comments: (i) I note that version 3.2 was compiled with the -fp-model strict option which was not used on version 3.1. I would need to check myself but it seems likely to me that this would potentially limit the optimizations the compiler is allowed to perform at -O2. I am curious if the authors think this might be significant for their apparent bug in the river runoff code? (ii) The Mare-Nostrum experiment of 3.2 uses -fp-model precise rather than -fp-model strict. Is this a typo or was it different to the CCA experiment? If so, why? (iii) Can the authors confirm these compiler options were applied to all the code? It is not uncommon to see different compiler flags on selected routines, or compiler directives in the code itself.

Page 12, line 5. It is disappointing that the authors have submitted this paper without completing their investigation into the cause of the disprecancy. If, in the time taken for the paper reviews, the authors are certain the fortran array referred to is the problem, this text should be amended. However, if there has not been any further investigation I would prefer not to see (educated) guesswork on the cause of the problem in the published paper and suggest removing the sentence, as it may turn out to be incorrect. The authors note that version 3.2 did not show the same behaviour. Does this mean that the code they suspect was different between the two model versions? Can the authors clarify in the text whether the offending code was different between the versions?

Corrections

Page 2, Line 29: 'reproducible' should be in italics to match 'replicable' and 'repeatable'.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2019-91, 2019.