

Interactive comment on “Devito (v3.1.0): an embedded domain-specific language for finite differences and geophysical exploration” by Mathias Louboutin et al.

Mathias Louboutin et al.

mlouboutin3@gatech.edu

Received and published: 19 December 2018

We thank you for your review time and very useful review to make this paper impactful. We answered all the reviewer requests and discuss our choices on revision changes. Please find attached below response to the review and attached the revised manuscript with the updates in blue.

Major Issues:

1. Page 5: lines 10-15, the authors claim that “Devito is compatible with a wide range of tools available in the Python software stack”, but it is not clear if this can

Printer-friendly version

Discussion paper



- be done on-the-fly using the operator function as devito produces a low-level C99 code. In short how the data is managed between C and Python is not clear, if file I/O is used then how is it advantageous?
- The memory is managed by numpy. What we mean is that even though we generate low level c99 code, the arrays are all numpy array that are accessible and modifiable in Python at any time. If file I/O is used, it would be in python and read into a numpy array or written from the numpy array, not within the C code.
2. Page 6: lines 1-2, the authors claim that CSE is used as an optimisation technique, as this is based on SymPy's CSE the reference given from 2015 is obsolete as the CSE capabilities of SymPy has changed a lot after that. Also, specifically the authors should address the following
 - a. CSE of SymPy considers the function arguments as sub-expressions, as all the operators are based on "Function" class how this is handled?
 - b. Also, the authors should specifically mention which version of SymPy is Devito compatible with

The Sympy reference is updated to most recent one, we changed it to the "cite us" for the sympy github repo. - a. Added sentence to precise that we do not use Sympy CSE but have a custom implementation of it. - b. Added version.

3. Page 6: line 10, "Devito provides two symbolic object types that mimic SymPy symbols, enabling the construction of stencil expressions in symbolic form", this statement needs justification as the next lines the authors claim to use "sympy function" which is not of type symbol, these two points contradict each other from SymPy point of view.

[Printer-friendly version](#)[Discussion paper](#)

- A symbolic expression is not necessarily a Symbol. While Sympy does the distinction between a Symbol and a Function, any Sympy expression is a symbolic expression (sympy = symbolic python).
4. Page 6: line 28, “TimeFunction” is it derived from “Function” class given in line 13 of page 6?, this should be clear in the manuscript
 - Added explanation of TimeFunction inheritance and extra parameters such as time_order for the time discretization and save for the size of the time axis if the full history is saved.
 5. As mentioned in the conclusions, such a framework can be applied to CFD problems. More examples should be given and I feel that only OpenMP parallelisation reduces the problem sets that can be solved for CFD, this should be clearly mentioned. Also, authors should provide a comment on the half-node interpolation capabilities of the framework as this is essential for most CFD cases
 - Added 3 CFD examples that highlight the flexibility of Devito. More example are available in the repository and this is now emphasized in the manuscript. The domain decomposition (MPI) and half-node FD are discussed.

Minor issues: 1. Type-setting fractions appearing in text should be inlined this should be implemented during type-setting stage

- Fixed

2. Page 16: line 3 it should be 10th and not 10th

- Fixed

3. Page 21: line 16 reference is not proper

- Fixed

4. The text in figures 16-18 are too small to read on print.

- Font size increased in figures for readability

5. Page 8: Figure 3 is confusing due to lack of borders

- Border added to all listings for consistency

6. The full form of the acronym FWI is repeated at two places this should be corrected.

- Fixed

[Printer-friendly version](#)

[Discussion paper](#)



7. Figures 16-18 How the operational intensity is evaluated?

- [Added explanation](#)

Please also note the supplement to this comment:

<https://www.geosci-model-dev-discuss.net/gmd-2018-189/gmd-2018-189-AC1-supplement.pdf>

Interactive comment on Geosci. Model Dev. Discuss., <https://doi.org/10.5194/gmd-2018-189>, 2018.

Printer-friendly version

Discussion paper

