Geosci. Model Dev. Discuss.,
doi:10.5194/gmd-2016-140-RC1, 2016

Geoscientific
Model Development
Discussions

Interactive comment on "A numbering algorithm for finite elements on extruded meshes which avoids the unstructured mesh penalty" *by* Gheorghe-Teodor Bercea et al.

Anonymous Referee #1

Received and published: 6 July 2016

The paper addresses the problem of generating sequential orders on mesh entities in so-called extruded meshes, with the goal to improve efficiency of memory access in finite element simulation software. Extruded meshes are defined as resulting from a tensor product of an unstructured mesh and a structured mesh - the paper particularly addresses layered meshes, which are unstructured 2D meshes in the horizontal and structured 1D meshes in the vertical direction (as often found in atmospheric/geoscience models). The authors present their implementation in the Firedrake software, and execute a careful analysis of achieved memory throughput for various low-order discretisation methods. Depending on the numbering of the base mesh, close-to-memory-bound performance is achieved once a certain number of layers is

exceeded.

General questions and comments:

- Data structures for layered meshes have been considered and implemented before - certainly in single-purpose codes, but also in frameworks (DUNE's prismgrid module, e.g.); I am aware that providing a survey of respective approaches to grid numbering in such packages might be impossible to do, but I think a general discussion on what options actually exist (and what implications resp. choices might have) when designing the numbering scheme could make the paper stronger.
- This is a bit related to the choice of title: at first reading I found myself expecting such a discussion; however, the paper clearly focuses on the approach followed in Firedrake (which is fine in itself, but a bit in contrast to the generic title and the abstract).
- You chose to number the DOFs of an entity contiguously, such that all DOFs of an edge (or cell) would be contiguous in memory. However, for low order methods and when the key design goal is to allow vectorization, you might want to strictly keep a stride-1 access on corresponding DOFs in layers - effectively this would mean exchanging the l and d_2 loops in Alg. 1. In any case, this choice depends on the type of operations we expect in simulations (whether we are strongly memory or compute bound, what the memory access patterns are, etc.), so a discussion on this would be interesting.
- As far as I got it, your concept of a stencil goes beyond the strict notion typically used for finite difference methods on structured grids: your stencils may also include element-local operations in finite-element- type methods (requiring a cell and its faces, edges, vertices) or also a face-based flux operation as in finite volume methods (which might require a face and its two adjacent cells). In any

case, you might explain this in a bit more detail, and maybe state one or two examples.

- What kind of unstructured mesh did you actually use for your results? You discuss in the paper that having a structured mesh as base mesh is advantageous for performance. Hence you might even explicitly address this issue by comparing results for a structured mesh (stored in an unstructured way) and one (or more?) typical unstructured meshes from applications.

Suggestions for improving the paper:

- As my only major suggestion, I would like to encourage you to switch from GFlop/s to GB/s in all performance plots: as you are in a memory-bound regime and the numbering scheme primarily addresses achievable "valuable bandwidth", "GB/s" would be the natural metric.
- You might check whether having a log-scale for x-axes makes the results for few layers better visible
- It would be helpful to a add a sketch for illustration of the indexing scheme defined in Eq. (5)
- I was wondering what kind of stencil a DG0xDG0 discretization would produce for the residual; aren't all accesses element-local then? In general, would it make sense to add a table (or similar) that describes which entities are accessed for the various discretisations?

Typos:

- p. 5, line 17: becoems -> becomes

- in the references, line 13, it should be Günther and Pögl (with umlauts)

Interactive comment on Geosci. Model Dev. Discuss., doi:10.5194/gmd-2016-140, 2016.