Geosci. Model Dev. Discuss., 6, C1230–C1231, 2013
www.geosci-model-dev-discuss.net/6/C1230/2013/

Geoscientific Model Development
Discussions

Interactive comment on "CUDA-C implementation of the ADER-DG method for linear hyperbolic PDEs" *by* C. E. Castro et al.

Anonymous Referee #1

Received and published: 12 August 2013

The authors present a CUDA implementation of the arbitrarily high-order ADER-DG finite-element method for the solution of linear hyperbolic PDEs publish in prior work. The abstract promises a not quantified speedup factor, which is even weakened by attributing it to the particular benchmark problem.

The introduction does not highlight the particular relevance of the method or its particular implementation for the geosciences and misses highly relevant citations such as Kloeckner et.al. "Nodal discontinuous Galerkin methods on graphics processors" 2009. The derivation of the numerical method, mostly taken from prior work of the authors, is very detailed and rigorous.

However the following section on the numerical implementation, which one would expect to be the centrepiece of the paper, is short and sketchy. There are no relevant

references cited apart from the CUDA programming guide and the hardware description is imprecise and does not use terminology properly: no mention of the SIMT programming model, the memory hierarchy or memory access patterns (coalescing) and confusion of what a kernel is vs. a hardware threat. The very short description of the implemented algorithm and data layout does not provide enough detail to be able to decuce any novelty or compare it to existing methods. Only the distribution of degrees of freedom to threads and blocks of the grid is given in detail, but without describing how threads cooperate to load/store from/to memory. Why was the memory layout chosen this way? Does it allow coalesced loads/stores? Is it different from what is used in the CPU implementation? The supplementary material with the kernel implementations appears to be missing.

The results section describes the convergence tests used to verify the implementation in great detail, whereas the discussion of the obtained performance is marginalised to half a paragraph. The speedup seen is not very meaningful given the baseline of an unoptimised CPU implementation and is not even quantified. The observation that the GPU advantage decreases with increasing order of the method is not further discussed. The conclusion does not provide any further analysis of the measured performance either or any evidence if or why the chose implementation is efficient. It ends on the weak note that the "speedup strongly depends on how much effort was used to code it", which is not an objective or scientifically rigurous criterion.

Interactive comment on Geosci. Model Dev. Discuss., 6, 3743, 2013.