Geosci. Model Dev. Discuss., 5, C564–C566, 2012
www.geosci-model-dev-discuss.net/5/C564/2012/

Geoscientific
Model Development
Discussions

Interactive comment on "Porting marine ecosystem model spin-up using transport matrices to GPUs" *by* E. Siewertsen et al.

Anonymous Referee #1

Received and published: 17 August 2012

This paper presents the results of a study into the feasibility of using GPU accelerators to perform marine ecosystem spin up. Overall, I think this is a good paper and represents a useful data point for future researchers considering parallelisation strategies for their models.

I have a number of small issues with the motivation for a GPU port and the analysis of the results which the authors can hopefully clear up.

1. p2181 "matrix-vector multiplications are one of the numerical operations predestined for GPUs".

In the case of sparse matrices, I do not see that this is immediately obvious. A short sentence or two describing /why/ the authors expect a speedup here would be useful. Perhaps noting that spMV (sparse matrix vector multiply) is typically dominated by memory bandwith requirements and that the GPU has a higher peak memory bandwidth than the CPU. Although note comments below.

2. p2198 Results

2a. You meant that for each measurement 100 runs were performed, but only the mean is reported. Given that all this extra data is available, it would be nice to have some data showing that the mean timings are reasonable values to report (e.g. the standard deviation of the results is low, etc...)

2b. It's unclear to me if the CPU data discussed here are runs on a single core of the Xeon system, or MPI-parallel across 8 processes. I /think/ it's the former but some clarification would be good.

2c. Comparison of CPU and GPU timing results is a tricky thing. I think this section could be improved significantly if the authors would present some data showing how much of the peak performance of the CPU/GPU they are obtaining (both in terms of flop-rate and memory bandwidth utilisation). I say this partly because I am rather suspicious of the MatCopy/MatScale/MatAXPY timings on the CPU relative to the GPU. In both cases, I think these should be limited by the available memory bandwidth and so it's strange to me that on the CPU they take a proportionally much larger amount of the total run time.

The algorithm used to compute the source-minus-sink term is not clear to me, and thus it is hard to compare the timing differences for BGCStep on the CPU and GPU. In particular, for the non-linear coupled case, the speedup for the BGCStep is higher than the overall speedup. Do the authors have any explanation for this phenomenon?

3. p2200 Conclusions "In our computations the effect [speedup] was totally as expected (about 20)"

Where does this expected factor of 20 come from? I see no obvious reference to it earlier in the paper.

In summary of these issues. I think the analysis would be much improved with the addition of some model of the expected performance. This would also presumably provide a firmer motivation for a GPU port "our model says we should expect a performance improvement of around 20, that's worthwhile so let's do it".

Finally, some minor (mostly typographical) issues I spotted reading the manuscript:

p2180 line 15 exemplary should be exemplar p2182 line 4 oft should be of p2182 line 19 the final sentence of this paragraph appears to be incomplete p2184 line 8 date should probably be data p2184 line 10 should "operator A" be "operator L" ?

p2192/2193. I think the introductions to the thrust and cusp libraries probably don't need example code, maybe just point the reader in the direction of the online examples these code snippets are taken from. p2194 I think it's uninteresting to the reader to learn what format PETSc functions should conform to. So I feel like one could lose p2194 line 22 until p2195 line 5.

Interactive comment on Geosci. Model Dev. Discuss., 5, 2179, 2012.