Geoscientific

Model Development

Discussions

Open Access

[GMDD]

Interactive

comment

Interactive comment on "Parallelizing a serial code: open–source module, EZ Parallel 1.0, and geophysics examples" *by* Jason Louis Turner and Samuel N. Stechmann

Anonymous Referee #1

Received and published: 2 December 2020

This paper addresses the issue of parallelizing a serial code used in geophysical applications. An open-source module named EZ Parallel is produced for this purpose to automatically with minimum invasion and knowledge of the user parallelize a typical coding setup of geophysical code. The authors demonstrate a very good speed-up of two example codes relative to the amount of user work invested in parallelization. The authors used MPI to achieve this goal.

Mayor point:

In algorithmic and modelling terms, the idea for this paper is not a new thing, but this paper with the supplemented open-source module does bring a merit for giving us a

[Printer-friendly version](#)

[Discussion paper](#)

somewhat simpler way of speeding up a user code on distributed systems, at least for very simple grids. I do like the fact how more advanced usage of MPI becomes hidden in the user code. However, I am concerned about the fact that the overhead of implementing MPI on the user side for simple grids is not that big compared to using EZ Parallel in the code. I would like to be explained more why using EZ Parallel is much easier than MPI for simple grids which you demonstrated. I think this paper would benefit more in explaining the tedious overhead of using MPI over EZ Parallel, in terms of additional lines of code and specific knowledge of MPI needed. Also assume we have a user who just coded two-level quasi-geostropic equation and has a pretty good idea how to use MPI to parallelise it. Why EZ Parallel is much better for him?

General questions:

1. Have you investigated using asynchronous MPI? Is there any problem adding support for OpenMP through EZ Parallel?

2. After parallelising the code and its output, one is faced with multiple output_procID which need to merged and visualized. Can EZ Parallel provide a tool to that, or is merging left to the user?

Minor remarks:

line 78: "with modification in red" but I do not see red text in Algorithm 2

line 195: andf -> and

line 211: Did you test scaling with I/O?

line 257: psuedo -> pseudo

Algorithm 4 after INITIALIZE_GRID: paralllelization -> parallelization

Fig 6 & Fig 7 take a large portion but are not directly relevant, they could be grouped together in Fig 6a 6b.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2020-257, 2020.