Geoscientific Model Development Discussions

Geosci. Model Dev. Discuss., author comment AC3
https://doi.org/10.5194/gmd-2021-320-AC3, 2021

Reply on RC2

Gaston Irrmann et al.

Author comment on "Improving ocean modeling software NEMO 4.0 benchmarking and communication efficiency" by Gaston Irrmann et al., Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2021-320-AC3, 2021

Thank you for your review and comments. We have changed the manuscript as to address your observations.

As you noticed, the domain decomposition, described in this paper, does not directly take into account the load increase due to the special halo exchange for the north pole. We have nevertheless introduced an optimization for this north pole issue that was not described in the original version of our manuscript. We added to the paper a detailed explanation of what we do that has been summarised in the following text :

When we look for the optimal domain decomposition, the choice of the number of lines and columns used to split the domain into the MPI subdomains we test, is done without considering the possible existence of the north pole folding in the configuration chosen by the user.
As we saw, the Euclidean division used to define the MPI subdomain may have a non-zero remainder (r), which forces some MPI subdomains to have one more point in the i/j directions than the other ones. This is where we added our optimization for configurations requiring the north pole folding.
Instead of cutting the domain with the following the following default j-decomposition: r □ jpj_{max} + (jpnj-r)□(jpj_{max}-1), we use the following j-decomposition: (jpnj-1)□jpj_{max} + jpj_{north} with jpj_{north} = jpj_{max} - jpnj + r.
We hence reduce the size of the northernmost MPI subdomains and improve the load balance.

To investigate the issue regarding abnormally slow time steps we had to break down the timing much further.
Indeed, the slowdown originates on a single core of the simulation at a very fine time scale. Because of the communications, by the end of the time step the delay produced by a slower core will be propagated to a significant portion of the simulation. We hence timed each call to an MPI routine inside NEMO's communication routine and the timed the execution in between MPI calls.

On the figure A attached, on each plot the horizontal axis is the index of the call to the communication routine in the time step, on the vertical axis is the execution time on a logarithmic scale. The different figures correspond to : (top, left to right) the East-West

communication, small computations following that communication, the North-South communication, (bottom, left to right) small computations and deallocation following the North-South communication, the computations following the whole communication (the code outside the communication routine) and the total time spent in the communication routine. For each figure the percentiles computed over the course of a simulation of 2080 time steps are plotted in colors and the average in bold black. In this figure, the measurement is done on the 39^{th} allocated core (MPI rank), the positions of the core is highlighted in the top middle graph. This core was chosen at random but all the cores of the simulation yield very similar results.

East-West communications are intra-node and do not involve interconnect network. As even for intra-node communications the maximum of the execution time is very high, the location of the nodes even on distant parts of the machine is not the cause of the slowdowns. The cores responsible for the slowdowns are also seemingly random as each core seems to behave in a similar way. As the slowdown occurs on random cores it is likely to be caused by the hardware or OS interference. We have tried on several supercomputers and still found this issue, occurring more often on some than others. We are currently investigating ways to make NEMO more resilient to such slowdowns.

The article was revised very concisely by adding the following sentence to the part about outliers : "A finer analysis showed that the slowdowns occur on random cores of the simulation."

Please also note the supplement to this comment:
https://gmd.copernicus.org/preprints/gmd-2021-320/gmd-2021-320-AC3-supplement.pdf