RC2:

Many thanks for the useful review and the additional technical information on the ocean component of E3SM that is being developed. It will be very interesting to see how your work there progresses. We have added this information to the text.

The reviewer comments:

The AI/ML revolution is causing much greater uncertainty, as it may displace the numerical methods we've used for 70 years. I think this point deserves a few more sentences in the text.

This is a good suggestion. We've added some text about this issue.

To respond to the remaining points:

L43: We have updated the Top500 reference (and associated text) to use the latest list as of Nov 2024;

L59: Thanks for pointing out that we'd missed describing inter-processor communication and the additional complexities this brings for GPUs. We've added a paragraph explaining the need for this communication and the implications for GPU.

L98-112: We've added a little more detail here.

The Strauss et al. reference has been updated (thanks).

–

RC1:

Many thanks for the review. Below we address the reviewer's comments and suggestions (reviewer's comments in red, our response in black):

This difficulty - and possible solutions - should be discussed in the paper and would increase greatly the value of the contribution. These solutions range from using software other than MPI, defining a new interface (with middleware) between FORTRAN codes and parallel/distributed processing at both node and processor levels or revising the FORTRAN codes. Developments in processor technology may introduce other opportunities beyond the current generation of GPUs.

We concur with the reviewer regarding challenges and solutions, however we think that the manuscript does describe various solutions, i.e., we address the various ways in which OOFS can make use of GPUs, ranging from the addition of OpenACC/OpenMP directives to existing

Fortran through to complete re-writes in C++, Python and Julia. We are unsure in what way we should extend the discussion.

Also, it would appear that the problem is being addressed by OpenMPI and that they have a version of MPI that takes advantage of GPUs.

As you say, GPU-aware MPI implementations are important and we have extended the text to make this clear.