Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2020-314-SC3, 2021

Geoscientific Model Development Discussions
Open Access
EGU

Interactive comment on "Parallel computing efficiency of SWAN" *by* Christo Rautenbach et al.

Christo Rautenbach

rautenbachchristo@gmail.com

Received and published: 12 January 2021

Author response: Thank you very much for the very useful and insightful comments. It is much appreciated. Please find the inline responses below:

1) The notion "thread" is a bit confusing for the MPI adepts, which should be "core". May be a combi "thread/core" would be a better wording.

Author response: All references to "thread" have been replaced with "thread/ core". 2) In general, within a single node (containing a number of threads/cores) OpenMP is more efficient than MPI. So, contrary to the study of Genseberger and Donner (2015, 2020), the results of your study contradict this general statement. Do you have an explanation for this? Perhaps, you may add some technical info concerning the used hardware with respect to this aspect (memory I/O, network, etc.)

Printer-friendly version

Discussion paper

Author response: OMP performs better at small thread numbers where MPI does better at larger thread numbers. This is illustrated in Figure 1 (a) and (b). This was clarified in the text on line 160 to 165. In Genseberger they note: “So, for this hardware the OpenMP version is twice as efficient as the MPI version.” It must be related to differences in the hardware. Details regarding the memory and network has been added in Methodology Section. They also looked at the Wadden Sea while this is a southern African benchmarking study. 3) One of the possible reason why a perfect speed-up cannot be obtained (see Fig. 1) is the domain partition of the computational grid and also the wet/dry (or active/inactive) points. The employed partition is the stripwise one which is because of the underlying parallel technique, namely the wavefront method. See Genseberger and Donner (2015) and also Zijlema (2005). The stripwise partition might not be the most optimal one with respect to the speed-up. In this specific case, it leads to a maximum of 6 threads/cores without too much sacrificing parallel efficiency. It would be good to highlight this aspect. Added reference: M. Zijlema. Parallelization of a nearshore wind wave model for distributed memory architectures. In Parallel Computational Fluid Dynamics - Multidisciplinary applications, pages 207-214. Elsevier Science, 2005. Do you have active/non-active grid points in your model schematization? Can you comment on this? Technical corrections, etc. line 103: ration -> ratio line 108: compliers -> compilers line 156: (16 x 25 threads) -> (16 x 24 threads) line 157: (16 x 24 threads) -> (64 x 24 threads)

Author response: Thank you for this great recommendation. These details and references have been added on lines 170 onwards, at the end of the Discussion section. Yes, the African continent computational points are inactive. This details has also been added in the Discussion section. All these corrections were made, thank you.

line 234: please change version number; also suggested to add the Technical Manual of SWAN besides the User Manual, as it contains the details of both physics and numerics Author response: Corrections made, and references added.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2020-314, 2020.

Printer-friendly version

Discussion paper