Comment on egusphere-2022-35
Anonymous Referee #1

Referee comment on "Block-structured, equal-workload, multi-grid-nesting interface for the Boussinesq wave model FUNWAVE-TVD (Total Variation Diminishing)" by Young-Kwang Choi et al., EGUsphere, https://doi.org/10.5194/egusphere-2022-35-RC1, 2022

The manuscript describes an extension of a popular wave model with the aim to increase applicability and computational speed through multi-grid nesting and MPI parallelization. The paper is well-written and most sections are properly explained. However, there are still a few sections, where more information is necessary. This applies mostly to memory sharing, synchronization and several other small details, which could really help a less experienced researcher in understanding the details of this presented technique. Below are a few comments, which can help to improve the overall quality and message of this paper.

Page 4 line 101-104:

"is 4 or better" meaning refinement factor >=4 or 4<=?

Page 7 line 176-183: Any particular reason why the spherical mode solve the weakly nonlinear equations and not the fully nonlinear equations?

Page 10 line 225-228: Is it necessary to exchange the dispersive terms: how does it increase the model efficiency.

How does the exchange work for the tridiagonal solver (child grid)?
Page 12 line 275-276: The restriction operator (The update operator) is not detailed. Which parent grid cells are actually updated? What about the ghost cells? What variables are updated in the Parent grid? Free surface/ velocity/ dispersion terms?

Page 13 line 293-295: This statement is not clear.

What terms are pre-computed before the model run?

The child-parent proximity (if proximity means boundaries) changes at each Parent time step – Same for the restriction process.

Personally I think the implementation is not very detailed considering that it’s the main contribution from the paper.

I dont understand how the communication between the child and the parent grid is straightforward...

When the authors talk about shared array allocation, do all processors have a copy of all the model variables (Parents and child grids) + the boundary condition of child grid? How does the synchronization work?

How is the MPI implementation optimized for nested grid?

How do the authors synchronize the parent and child grid after each parent time step? Do the authors use a fixed time step for the Child grid? If they use the CFL condition for the parent and the child solutions, does this involve that some type of synchronization is required before the update step.
Application

4.1 Evolution of an initial rectangular-shaped hump

Symmetry test is okay

Figure 7: maybe include a diagonal transect and plot (a) (b) and (c) in the same figure. For better comparison.

Page18 line 349-352: The whole solution depends on grid resolution not only dispersive effects.