Review of "Parallel level-set methods on adaptive tree-based grids." Submitted to Journal of Computational Physics, Authors Mirzadeh, Guittet, Burstedde, and Gibou.

In this article, the authors extended the "p4est" oct-tree parallel libraries in order to accelerate the computation of dendrite growth simulations.

I like this article. The authors' algorithm distributes a local grid structure to all the processors, overcomes the CFL constraint, and iterates so that the new grid G^{n+1} adapts to the new interface. I accept this paper pending the following changes:

- It took me a while to understand Algorithms 1 and 2. It is recommended that the authors' pseudocode be closer to a language that many people are familiar with like C++ or Fortran 90 (or perhaps, the authors' are using a new language that this referee is not aware of?).

- In Algorithm 2, what is the definition of "rank"? "owner's rank"? "mpirank"? "st"?

- In algorithm 2, if X is contained within the mesh on a given processor, does that mean all the points needed in order to derive the value at X are also on the same processor? What if the points in the interpolatory stencil of X belong to multiple processors? Which processor does the actual interpolation? Is it the processor that needs the value at X? or is it the processor that owns X?

- In section 4.2, the authors should do a standard 3D reversible test when checking speed-up. A standard test allows researchers to compare the new method to previous schemes. Also the authors should report the numerical error (e.g. the symmetric difference error or the level set error in proximity to the zero level set). For a reversible problem, the exact solution is known at the end time. So in Table 2, an extra column should be added to both the (a) and (b) parts which has the error (it is assumed that the error is independent of p?). For the reversible problem, the zero level set isosurface should be displayed at maximum deformation $t = T/2$ and at the end time $t = T$ for some representative values of "CFL" and l_{max}.

- Also in section 4.2, how frequently is reinitialization carried out? What is the pseudo time step $\Delta\tau$? What is τ_{stop}? If there is no reinitialization

for this test, it is requested that the semi-Lagrangian algorithm together with reinitialization be analyzed in this section too.