Geoscientific Model Development Discussions

Geosci. Model Dev. Discuss., referee comment RC1
https://doi.org/10.5194/gmd-2021-233-RC1, 2021

Comment on gmd-2021-233

Anonymous Referee #1

Referee comment on "LAPS v1.0.0: Lagrangian Advection of Particles at Sea, a Matlab program to simulate the displacement of particles in the ocean" by Maxime Mouyen et al., Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2021-233-RC1, 2021

Review of "LAPS v1.0.0: Lagrangian Advection of Particles at Sea, a Matlab program to simulate the displacement of particles in the ocean" by Mouyen et al

In this manuscript, the authors present a Matlab-based toolbox for integrating virtual particles in three-dimensional ECCO2 fields, optionally also with Stokes drift from WaveWatchIII. They show the fidelity of this code through a few examples related to tracking of sediments of the Amazon plume and plastics in the northwest Pacific Ocean.

While I strongly support the development of new Lagrangian oceanography codes, since that increases diversity and hence facilitates model comparison, I must say that I am not sure that LAPS provides sufficient unique and scientifically novel features to warrant publication in a journal like GMD. Perhaps the Journal of Open Source Software (https://joss.theoj.org/) would be a better outlet?

In particular, LAPS seems to have no added features beyond already existing codes. The authors state that its Unique Selling Point is the ease-of-use, but then leave it to the users to e.g. download the ECCO2 data themselves. They also don't discuss any tools for the analysis and plotting of the trajectory files; which is where the largest intellectual/technical challenges are for most new users. So I don't think that users without experience with running Lagrangian software can simply pick up this tool and work with it; and if they do have experience with coding then LAPS does not add much to the already-existing ecosystem of Lagrangian tools.

Beyond this fundamental problem of novelty; I also have nine major comments on the manuscript itself:
1. It is unclear whether ECCO2 and WaveWatch are in fact consistent products. Figure 1

suggests that the Stokes drift does not leave any imprint in the currents, which is strange. Are they forced by the same data products? If not, what is the impact of that?
2. There is no mention at all on the type of interpolation schemes used. Are they linear? Spline? What about temporal interpolation? How is beaching treated?
3. Algorithm 1 suggests that the timestepping scheme is simple Euler Forward? This is known to be very inaccurate. Have the authors tested this?
4. The WaveWatchIII data is interpolated to the finer ECCO2 data as this reduces interpolation time; but the authors do not mention that of course this increases storage and/or memory.
5. The unit-testcase in lines 152-160 is extremely trivial, and does not test most of the interpolation and integration schemes. I strongly encourage more thorough testing with more complex (analytical) flows
6. The discussion on depth-dependence of Stokes drift is fairly simplistic. How about the return flow discussed in e.g. Van den Bremer and Breivik? Can that be incorporated too?
7. The runtime tests are only done for very small particle umbers. Meaningful Lagrangian oceanography experiments use at least 10^5 or 10^6 particles, which is 100-1000 times larger than the tests here. How scalable is the code to these numbers of particles?
8. The assessment of model skill in Figure 4 is rather meaningless. First of all, it tests the fidelity of the ECCO2 flow fields, more than the LAPS code. And secondly, the statement that the errors are 'reasonable' (line 181) is rather meaningless. Why are these standard deviations reasonable? How did the authors decide? It would have been much stronger to have an independent criterium for accepting the fidelity
9. The analyses in Figures 7 and 8 are done with an extremely low number of particles, so there is no statistical significance at all to these results. I strongly recommend to either do these analyses much more carefully, or not at all.

Smaller comments
- line 43: Are these particles infinitesimally small? Or do they have some inertia too?
- line 49: is the 15-minute timestep hardcoded? How is it chosen? Some kind of CFL-criterium? Would it not depend also on the domain?
- line 56: Does the algorithm deal with periodic boundaries too (i.e. if the particles occupy a small region straddled by the zonal domain boundary in ECCO2)?
- line 57: Why choose 6 degrees halo? Where does this choice come from? Is it hardcoded?
- line 98: why are these values for kappa and z_0 chosen? Are they hardcoded?
- Fig 2: Would it make sense to flip the y-axis so that (negative) depth is downward?
- line 110: Where does this equation 4 come from? This is only valid for certain range of D, right? Is this checked in the programme, so that users don't run with too large values of D?
- Table 1: I don't really see the relevance of this table for the manuscript. Leave out?
- line 111: Why is \rho_s a constant? Why not compute it directly from the ECCO2 temperature and salinity fields? That would be much more accurate!
- line 119: How is this probability defined? Is it spatially variable?
- line 143: Do I understand this statement correctly that the full particle trajectories are stored in memory?? How scalable is this, for millions of particles and thousands of time steps?
- line 168: The assumption that a drogued drifter drifts with the velocities at 15m is too simplistic. In reality, it integrates the velocities over the extent of the drogue. This would be a much better way to simulate drogued drifters