Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2020-154-AC1, 2020
© Author(s) 2020. This work is distributed under
the Creative Commons Attribution 4.0 License.

Geoscientific Model Development Discussions
Open Access

Interactive comment on "Numerical integrators for Lagrangian oceanography" *by* Tor Nordam and Rodrigo Duran

Tor Nordam and Rodrigo Duran

tor.nordam@sintef.no

Received and published: 23 July 2020

Dear Anonymous Reviewer #1,

Thank you for your comments. We are glad you agree with the motivation for the manuscript, and that you say it would be a welcome addition to the field.

Regarding your comments, we address these below, and we also outline the changes we suggest to make in the manuscript.

Major comments

> 1) The authors make the case for higher-order spatial interpolation, as if that is always better. However, there is no discussion at all about how the order of spatial interpolation is related to the order of the advection schemes within the OGCM from which the data is derived. Intuitively, I'd say that the most appropriate interpolation would be the scheme that most closely mimics the model advection scheme. I'd suggest the authors discuss this.

There are two points in this comment that we would like to address.

First, it is not our intention to advocate higher-order interpolation as always better. There are advantages and disadvantages to all types of interpolation. Among the disadvantages of linear interpolation are the obviously unphysical discontinuous derivatives. Among the disadvantages of higher-degree splines is the increased computational effort per evaluation, and the possibility of overshoot/oscillations. The question we seek to address in this paper is of a numerical nature: "Given an interpolation scheme, which integrator gives the best balance between accuracy and performance?". We touch upon this in lines 333-335, but we agree that this can be made more clear in the manuscript. Towards the end of the introduction, around line 65 in the original manuscript, we will add the following (or something very similar):

"The purpose of our investigation is not to investigate how well different model resolutions and different interpolation schemes reproduce physical drifter trajectories. Rather, we address the purely numerical question of which combinations of integrator and interpolator give the best work–precision balance, for a given resolution."

Second, we feel that the point about using an advection scheme that mimics the ocean model is outside the scope of the paper. Our motivation is to provide guidance towards solving the common problem of integrating trajectories from offline velocity fields. Being a common oceanographic task, diverse ocean models are

CC BY

used, and each ocean model often has different advection schemes to choose from when configuring the model. For example, a popular ocean model, ROMS, offers four horizontal advection schemes for the user to choose from: second-order centered, fourth-order centered, fourth-order Akima and third-order upwind; e.g. see https://www.myroms.org/wiki/Numerical_Solution_Technique#Horizontal_and_Vertical_Advection.

Advection schemes in ocean modelling is also a topic of active research, and new methods are expected to be introduced. See for example

https://www.sciencedirect.com/science/article/abs/pii/S146350031000106X
https://www.sciencedirect.com/science/article/abs/pii/S1463500311001831
https://www.sciencedirect.com/science/article/abs/pii/S1463500308001510
https://archimer.ifremer.fr/doc/00435/54690/

Additionally, information about the advection scheme is not always readily available for public ocean current data sets. It is therefore unpractical to try to mimic an advection scheme for the general problem of interest studied here. Our intent is to help oceanographers implement an efficient method to integrate any ocean model velocity provided on a rectangular grid, without further concern.

> 2) The authors also but then quickly step over the problem of interpolation near land. Higher-order interpolation would mean that the halo of land is further extended into the ocean (as very briefly mentioned in line 362). I feel that this should be given more discussion. How would this problem compare to the error that the 'consistency of order p of the numerical method is no longer satisfied when the derivatives are not continuous' (lines 222-223)

It is worth commenting on this issue, but a full discussion would, we feel, be outside the scope of the current study. The global integration error is a universal issue that occur in all Lagrangian models using numerical integration of ODEs, while the handling of land is very much application dependent. For applications like LCS calculations, and

CC BY

global or large-scale transport simulations, interaction with the coastline may be almost negligible, whereas for applications like near-shore oil spills coastline interaction may be very important, and is implemented differently in different oil spill models (see, e.g., https://www.mdpi.com/2077-1312/6/3/104). Hence, it is impossible to say something generally applicable about how the error due to the handling of land cells compares to the global interpolation error.

We will add the following at line 364:

"Note that with higher-degree interpolation schemes, the fact that we set the currents to zero in land cells will have an effect on one or more of the closest cells to the coastline. For applications such as oil spill modelling, where shoreline interactions are important, this may be a disadvantage."

> 3) There is no discussion at all about the widely-used Analytical advection scheme of e.g. https://doi.org/10.5194/gmd-10-1733-2017. How does that scheme compare to the integrators discussed here?

Implementing a Lagrangian particle tracker with an interpolation scheme and an ODE solver can be accomplished very quickly, with just a few tens of lines of code in, e.g., Python or Matlab. As evidenced by the literature, many authors implement their own Lagrangian particle tracking codes in this way, and use different combinations of interpolation and integration schemes. Hence, we feel that the current manuscript provides useful information to the community, even without a detailed discussion/comparison to more advanced schemes.

While we are not very familiar with the advection scheme used in TRACMASS, it is our understanding that it (bi- or tri-) linearly interpolates the current internally in each cell based on vector components at the cell faces (?), and then solves analytically to find the passage of a particle through the interpolated field inside a cell. Rather than using

CC BY

a specified timestep or tolerance, this approach is more "event driven", calculating the trajectory through a cell as one step.

In the case of trilinear interpolation, it should be possible to compare our trajectories to those obtained with TRACMASS, but for the higher-degree interpolation schemes a direct comparison would be (numerically) meaningless. In any case, we feel that a detailed discussion/comparison to the TRACMASS trajectory scheme would be outside the scope of the current investigation, but might be an interesting topic for a future study.

> 4) The authors mention that they ignore diffusion (line 74). However, in most applications diffusion will be included in the computations. I wonder whether the errors caused by the finite-sized set in the Wiener process are not much larger than any errors in interpolation as discussed here. It would be good if the authors could comment on this.

This is not entirely straightforward. Strictly speaking, advection-diffusion problems are not modelled with ODE methods, but with SDE methods, which are usually of low order. While it might be common in practice to use a higher-order ODE method, and tack on a random displacement in an ad hoc manner, such a splitting of the problem is in itself an approximation which introduces an error. For spatially varying diffusivity, the short timestep required for the SDE method to give a sufficiently small error (in the weak sense) may also render the advection error irrelevant. A thorough discussion of this issue is definitely outside the scope of the paper.

However, it is of course clear that adding random increments to the position of a particle may in many cases dominate the numerical integration errors. In these cases, it would probably give little or no extra benefit to use higher-order integration schemes. We propose to add the following paragraph towards the end of the Discussion (probably in Section 5.3):

Printer-friendly version

Discussion paper

"As mentioned in Section 2, we have considered pure advection, ignoring diffusion. However, for many applications diffusion must be included. Solving the advection-diffusion equation with a particle method amounts to numerical solution of a stochastic differential equation (SDE), instead of an ODE. A range of different SDE schemes exist, and the details differ, but all such schemes involve adding a random increment at each timestep. If the random increment is far larger than the local numerical error in each step, then the numerical error in the advection is probably of limited practical importance. The details will depend on the application, and we encourage experimentation. A detailed description of numerical SDE schemes is outside the scope of this study, but the interested reader may find it useful to refer to, e.g., Kloeden Platen (2013), Gräwe et al. (2012) and Spivakovskaya et al. (2007)."

References:

Gräwe, U., Deleersnijder, E., Shah, S. H. A. M., Heemink, A. W. "Why the Euler scheme in particle tracking is not enough: the shallow-sea pycnocline test case." Ocean Dynamics, 62(4) (2012) pp. 501-514.

Kloeden, Peter E., and Eckhard Platen. Numerical solution of stochastic differential equations. Vol. 23. Springer Science Business Media, 2013.

Spivakovskaya, Darya, Arnold W. Heemink, and Eric Deleersnijder. "Lagrangian modelling of multi-dimensional advection-diffusion with space-varying diffusivities: theory and idealized test cases." Ocean Dynamics 57(3) (2007) pp. 189-203.

> 5) I wonder why the authors don't test their method on (complex) flows where an exact solution is known. Quite a few of these flows have been used in the literature, including e.g. the Bickley Jet. That would save them a lot of challenges in defining the 'exact' solution.

An important motivation for avoiding an analytical reference solution, is that it seems

CC BY

likely that fifth-degree spline interpolation would out-perform cubic splines and linear interpolation in terms of accuracy, when comparing to an analytical solution. That would lead to the conclusion that higher-degree splines give more accurate results, which is by no means certain for ocean currents. By using numerically obtained reference solutions, obtained separately for each interpolation scheme, we remain "interpolation-agnostic", merely addressing the question of which integrator/interpolator pair is numerically more efficient.

As mentioned in our response to comment 1, our intent is to help oceanographers implement an efficient method to integrate any ocean model velocity. Hence, we wanted the discussion and conclusions to feel directly relevant to applied oceanographers. For example, we conclude that the most efficient numerical approach for a given level of accuracy depends on the spatial resolution of the dataset. If we were to start with an analytically defined flowfield, and then discretely evaluate this on different grids, for later interpolation, it is not obvious how any conclusions could be applied directly to ocean current datasets.

> 6) I am not sure if comparing ends points is the most appropriate metric. Why not compare the along-trajectory differences, which is commonly used in the field (e.g. https://doi.org/10.1029/2018JC014813), so that the full trajectories are taken into account.

The endpoints were chosen simply because of the discussion in terms of the theory for numerical solution of ODEs. The standard approach in the ODE literature is to discuss convergence in terms of the global error, the order of a method refers to the global error, etc. In Figure 4, the error as a function of number of evaluations for the fixed-step integrators make straight lines in the log-log plot, with a slope determined by the order of convergence. If we considered the along-trajectory error, there would be less of a direct link to the theory of ODE methods when discussing these results.

CC BY

From an application point of view, different metrics could be appropriate. For, e.g, LCS applications, only the endpoint matters, while for other applications the entire particle history might be relevant. In light of these points, we feel that the end points are, on the whole, the most appropriate metric.

> 7) The authors spend a large amount of attention on their 'special-purpose integrators' (section 3.3). However, they don't mention that most implementations of Lagrangian integrators would quite naturally implement such special-purpose integrators, simply because they don't store all time slices in memory so that they need to stop integration on the time of each time slice in order to load the next one.

We believe this is a misunderstanding. Taking the example of cubic interpolation, one needs at least 4 time slices in memory to construct the interpolation. In the case of our special-purpose integrators, integration is stopped and restarted at every one of those time slices, not just when new data must be loaded.

For linear interpolation, two time slices will suffice to construct the interpolator, but in for example Taylor and Shadden (2008), which uses linear basis function interpolation and a variable-step Runge-Kutta-Fehlberg method, there is no description of stopping and restarting integration at every time slice. In our opinion, it is often hard to be sure of the exact details of how the full trajectory calculation has been implemented, given the typically very short descriptions of interpolation and integration schemes in the applied literature. Hence, a thorough discussion should be of value to the community, even if the method itself were to have been used by others before.

For fixed-step integrators, we do discuss the fact that these perform very well when the timestep is chosen such that integration is stopped and restarted at every time slice. See, e.g., lines 415–420 in the Discussion, and 515–520 in the Conclusion.

Printer-friendly version

Discussion paper

CC BY

Minor comments

- line 4: clarify here that this is interpolation in space and time?

OK.

- line 16: ’computations on data from atmospheric models’?

OK.

- line 30: ’For all these applications’?

OK.

- line 36: ’capable’ is an anthropomorphism; hyperbolic points are not capable of anything.

Will rephrase.

- line 36: By whom is this recommended?

Agree that this is unclear, will add reference or rephrase.

- lines 43-54 and 126: It might be very useful to include a table with details of the different integrators, so that readers don’t need to dig into the literature themselves to find out what the specifics are of each of these

Printer-friendly version

Discussion paper

This comment refers to the introduction, where we mention different integrators used in the applied literature. Given that these include not only Runge-Kutta methods, but also linear multistep methods and predictor-corrector methods, a useful description of these would probably require several pages. We don't see that a table could contain enough information to be useful, and for anyone who wants to implement these methods, looking them up would not be a large amount of work.

> - line 55: 'very common' is perhaps too strong? Lagrangian oceanography is still a bit of a niche

Will drop "very".

> - line 93: why is x not bold here?

We have left x in non-bold for the general theory discussion. Will add a sentence to make this more clear.

> - line 104 and other places: Would errors e and E not always be absolute values?

No. The error it self can be either positive or negative (see, e.g., Eq. (3.1) in Hairer et al. (1993)). However, when we discuss how the error scales with the timestep, it's the absolute value.

> - line 172: At least summarise where these values 2.5 and 0.8 come from

This is a mistake on our part, as the reference given doesn't actually discuss the choice in detail. Will rephrase line 172 as follows:

CC BY

"The factors 0.8 and 2.5 were chosen from a range of values recommended by Hairer et al. (1993, p. 168), and were kept constant for all numerical experiments."

> - line 176: How does this extent to staggered grids (e.g. Arakawa-C) which are often used in oceanography?

Simply interpolate each vector component on its own grid. The vector components are interpolated separately, so this would require only a minimal change at the initialisation of the interpolators in our implementation.

> - line 191: The word 'quite' is somewhat vague here

OK, will drop the word "quite".

> - line 224: give some examples relevant to Lagrangian oceanography of these cases?

We propose to add the following at the end of line 225:

"The more pathological examples are probably not likely to occur in practice. However, as we will see later, the unbounded error in a single step can in some cases dominate the global error, making the use of a higher-order scheme pointless."

> - Figure 1: Also show (some of) the trajectories here, to give readers a feeling for the extent of dispersion?

We assume this comment is meant to refer to Figure 3? We have created a figure (attached) showing the final positions of the particles, for the three different datasets. We propose to add this figure either at the start of Section 5, or in an appendix.

Printer-friendly version

Discussion paper

CC BY

- line 320: it is unclear at what depth the particles are released, and also whether they are advected in 2D or in 3D

OK. It is stated in line 360 that we use only the surface layer of the currents, but this should be described clearly in Section 4.1 or 4.2 as well. We propose adding the following after line 300:

"We have chosen to consider two-dimensional (horizontal) transport only, using the surface layer of the modelled current data. The current velocity field is interpolated in three dimensions (two spatial dimensions plus time), using the same degree of interpolation in all three dimensions."

- line 329: is 'transport' the best word here?

We propose to change this to "trajectories".

- line 363: explain what kind of padding is done

Padding was probably not the right word. It only means that when cropping the dataset, the subset selected was larger than then minimal size required to cover all the trajectories. Will rephrase.

- line 373: explain why this creates additional discontinuities

This is somewhat technical, and hard to explain without going into details on how spline interpolations are constructed, which we have not covered in the paper. It is also a bit on the side, as we don't investigate this type of interpolator. Therefore, we would prefer not to go into detail, and simply refer the interested reader to Lekien and Marsden.

CC BY

- line 381: what is the standard deviation/variability around this median? Are the differences between the runs larger than the variability within the runs?

Not sure how to best show the standard deviation. The plots in Fig. 4 would become unreadable if we added error bars to every datapoint. (Also, recall that everything is completely deterministic. Therefore, the "variability within the runs" comes only from the different initial positions.)

An option might be to add this as supplementary information, either in the form of several figures, or a table.

- line 388, 389 and 390: even though the authors make a good point about comparing number of computations instead of runtime, here they still mention that runs are 'longer' and report runtime in seconds.

That is only meant to illustrate that each evaluation takes longer when higher-degree interpolation is used. We will make this more clear.

- Figure 4: I'm a bit confused by the number of points on each of these lines. Why do some have 11 points, even though according to table 1 there were only 9 time steps and 10 tolerances tested?

This is clearly an inconsistency between Table 1 and the figures. We will fix this.

- Table 2: Would this data be easier to parse in a figure instead of a table?

In our opinion, a table is most suited in this case.

Printer-friendly version

Discussion paper

- line 493: mention what is 'special' about these special-purpose integrators (e.g. 'that don't step across time grids' or something like that)

Ok, will expand/rephrase.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2020-154, 2020.

CC BY

Interactive comment

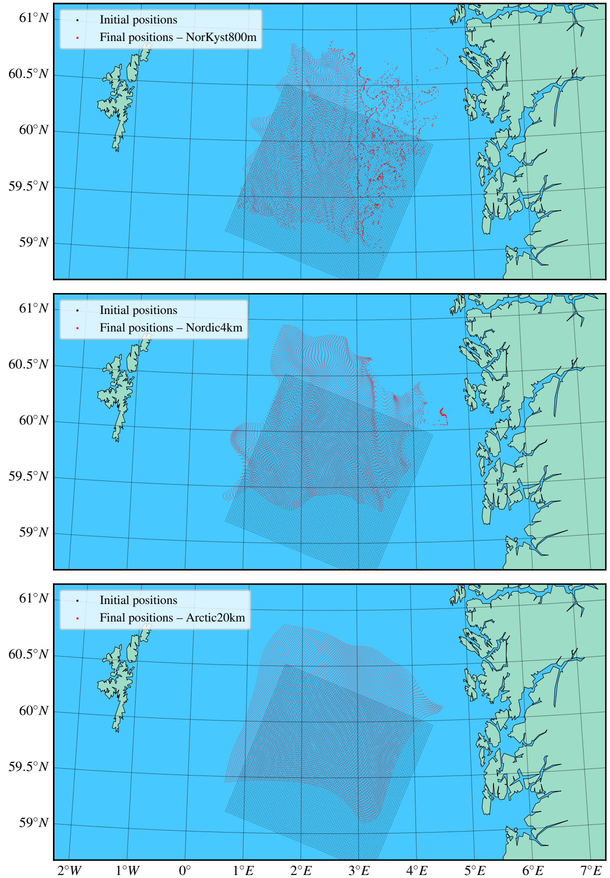

Fig. 1.

Printer-friendly version

Discussion paper

CC BY