Geoscientific
Model Development

Performance of offline passive tracer advection in the Regional Ocean Modeling System (ROMS; v3.6, revision 904)

Kristen M. Thyng[1], **Daijiro Kobashi**[1], **Veronica Ruiz-Xomchuk**[1], **Lixin Qu**[2], **Xu Chen**[3], **and Robert D. Hetland**[1,a]

[1]Department of Oceanography, Texas A&M University, College Station, TX, USA
[2]Department of Earth System Science, Stanford University, Stanford, CA, USA
[3]Center for Ocean-Atmospheric Prediction Studies, Florida State University, Tallahassee, FL, USA
[a]now at: Pacific Northwest National Laboratory, Richland, WA, USA

Correspondence: Kristen M. Thyng (kthyng@tamu.edu)

Abstract. Offline advection schemes allow for low-computational-cost simulations using existing model output. This study presents the approach and assessment for passive offline tracer advection within the Regional Ocean Modeling System (ROMS). An advantage of running the code within ROMS itself is consistency in the numerics on- and offline. We find that the offline tracer model is robust: after about 14 d of simulation (almost 60 units of time normalized by the advection timescale), the skill score comparing offline output to the online simulation using the `TS_U3HADVECTION` and `TS_C4VADVECTION` (third-order upstream horizontal advection and fourth-order centered vertical advection) tracer advection schemes is 99.6 % accurate for an offline time step 20 times larger than the online time step as well as online output saved with a period below the advection timescale. For the `MPDATA` tracer advection scheme, accuracy is more variable with the offline time step and forcing input frequency choices, but it is still over 99 % for many reasonable choices. Both schemes are conservative. Important factors for maintaining high offline accuracy are outputting from the online simulation often enough to resolve the advection timescale, forcing offline using realistic vertical salinity diffusivity values from the online simulation, and using double precision to save results.

1 Introduction

The ability to integrate Eulerian tracer fields offline or separate from the online original full simulation is attractive because of the improved computational efficiency. Once an online simulation has been run, any number of offline simulations can be run, forced by the stored online model output, using a larger time step, and only needing to integrate the transport field itself. This allows for many simulations when, in contrast, fewer would have been possible with the online simulation. This study presents the development and assessment of an offline passive tracer advection model that is part of the Regional Ocean Modeling System (ROMS), version 904, in the Coupled Ocean–Atmosphere–Wave–Sediment Transport (COAWST) modeling system (Shchepetkin and McWilliams, 2005; Warner et al., 2010). While the ultimate goal of this work is to run ROMS with both offline floats representing oil and tracers representing biological processes, along with sediment–oil interactions, the present focus is on the offline tracer model with a passive tracer.

Previous work has been done in this area with other models. An offline tracer model for the Massachusetts Institute of Technology (MIT) general circulation model (MITgcm) was developed and showed good accuracy (Hill et al., 2004). For a set output frequency, an offline time step of 8 times the online time step gave a skill score of over 98 %. An offline tracer model based on MITgcm has been used in several studies (Dutkiewicz et al., 2001; McKinley et al., 2004). Another offline tracer model, the offline global 3D ocean

tracer model (OFFTRAC), is based on the Hallberg Isopycnal Model (HIM) and has been used for long-term biogeochemical integration (Zhang et al., 2014). Other tracer models have been developed separately from a full numerical ocean simulator. Gillibrand and Herzfeld (2016) developed a separate tracer advection model that is not numerically limited by the Courant number as is expected in the present case. Another such model developed by Khatiwala et al. (2005) has a different approach entirely to offline tracer advection, using a mathematical approach that is distinct from more commonly used numerical tracer integration. Interestingly, Lévy et al. (2012) found that for particular dynamical scenarios, degrading online model output spatially can result in offline computational savings with little accuracy degradation.

The offline tracer model described in this paper is integrated into and derived from the ROMS model: preprocessor flag choices allow access to the offline capability. While not being derived from a specific ocean model allows for wider potential use, as in some of the previously described models, there may be an advantage in using the offline model that is a derivative of the offline model to ensure consistency, using the exact same numerics and setup. The expected user for this software is someone who uses ROMS for their ocean modeling needs and wants to have the ability to run more tracer simulations, decoupled from their more expensive online simulations. Another type of user may simply have some ROMS output available, and this code will allow them to leverage it beyond its originally intended use.

The experimental setup is described in Sect. 2; this includes a description of the model setup (Sect. 2.1), the offline experiments (Sect. 2.2), and the metrics used for evaluation (Sect. 2.3). Results are shown in Sect. 3, and a discussion of results is given in Sect. 4. Specific code descriptions are provided in the Appendix: code changes made for robust offline tracer advection (Appendix B) and a description of how to set up both online and offline simulations for best results (Appendix C).

2 Experimental setup

2.1 Online model setup

The online model is set up for the northern Gulf of Mexico (25.6–30.6° N, 94–84° W). The domain was chosen because the final goal of this work is to simulate the fate of oil spilled in this region in 2010. The horizontal resolution is 0.04° to fully resolve mesoscale processes, and there are 50 vertical layers with refinement at the seabed and sea surface (the transformation equation parameter `Vstretching` is 5, and the stretching function parameter `Vtransform` is 2) (Azevedo Correia de Souza et al., 2015). The time step is 20 s. Hybrid Coordinate Ocean Model (HYCOM) Global Reanalysis data (experiment "GLBu0.08/expt_19.1") are used to initialize the model and provide boundary condi-
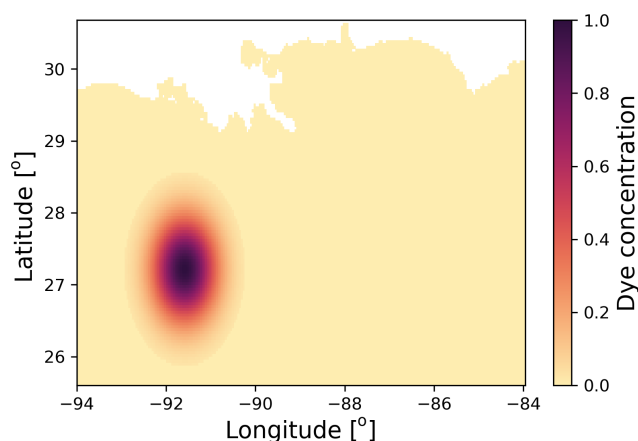

Figure 1. Model domain and initial blob of passive dye at the surface for the first set of numerical experiments.

tions (Fox et al., 2002; Cummings, 2005; Chassignet et al., 2007; Cummings and Smedstad, 2013). The surface forcing is provided by hourly Climate Forecast System Reanalysis (CFSR) data (Saha et al., 2010), and air–sea turbulent fluxes are calculated using bulk formula COARE 3.0 (Fairall et al., 2003). In order to realistically simulate the water properties and dynamics in the coastal area, 21 daily river discharges with specified water transport flux and temperature (from the United States Geological Survey) are implemented as boundary fluxes along the coast. To stabilize the open boundaries, lateral nudging layers are set at the open ocean boundaries. The nudging timescale is 0.04 d at the boundary and is gradually tuned to 10 d at the 18th interior grid. The climatology used for the nudging process is also provided by the HYCOM output. This online model ran for 90 d, from 20 April to 19 July 2010, but a subset of 14 d is used for the present experiments.

2.2 Offline experiments

2.2.1 Full water column Gaussian

A series of online and offline simulations were run to evaluate the comparative performance of offline tracer advection. The first set of numerical experiments presented in this paper were initialized with a discrete Gaussian blob of dye southwest of the Mississippi River delta in a regional model of the northern Gulf of Mexico (see Fig. 1). The blob of dye extended fully through the water column. Online simulations were run with two tracer advection schemes: `MPDATA` (both horizontal and vertical), and `TS_U3HADVECTION` and `TS_C4VADVECTION` (third-order upstream horizontal advection and fourth-order centered vertical advection, respectively, which is shortened to `U3C4` for the remainder of the paper). Additionally, the online simulations were output at different frequencies, as multiples of the time step (the

`nhis` and `navg` parameters for ROMS):

$nhis = 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000.$

Given that the time step of the online simulation was 20 s, these correspond to output frequencies of about 20 s, 40 s, 100 s, 3.3 min, 6.7 min, 16.7 min, 0.56 h, 1.1 h, 2.8 h, 5.6 h, 11 h, and 28 h, respectively. Online simulations were saved as both instantaneous snapshots (`his` files from ROMS) and as averages across time steps (`avg` files from ROMS).

These offline simulations were run using one of the two tracer advection schemes, with either `his` or `avg` files as climatology at the output frequency from the online simulations (controlled by `nhis`/`navg`). Additionally, they were optionally forced by the vertical salinity diffusion variable, `Aks`, as calculated by the online simulation or by just the background value. Finally, for an input climatology file from an online simulation of a given output frequency (`nhis`/`navg`), offline simulations were run with a time step from the list of `nhis` values of up to the same output frequency as the online simulation. A time step of 50 times the online time step was found to lead to unstable solutions; thus, in effect, the offline time step could be 1, 2, 5, 10, or 20 times the online time step, but it could never be larger than the `nhis` value for the given simulation. Also note that the offline time step needs to divide evenly into the output frequency in the climatology file so that only two time steps are being accessed at a time. Therefore, for a climatology forcing file of `nhis` = 50, the offline simulation could not be forced with `dt` = 20.

The relevant controlling timescale for this simulation is the advection timescale. Results from online simulations of the dye advection show a representative length scale of about $L = 10$ km and a speed of about $U = 0.5 \, \mathrm{m \, s^{-1}}$, giving an advection timescale of $T = L/U = 20\,000$ s, or about 5.6 h. This timescale is specific to the location of the dye patch, which is off the continental shelf and responding to mesoscale processes. If the dye patch was on the shelf, one would expect a shorter timescale. The timescale will be used to normalize times given in the results and to interpret accuracy in relation to offline time choices.

2.2.2 At-depth realistic Gaussian

Another set of simulations were run to apply the lessons learned in the first set to a more realistic test case (Fig. 2). This test case is meant to represent an infusion of some material to the ocean at depth, for example dissolved methane gas. However, as we are testing only the passive offline tracer advection scheme in the present study, the tracer is passive and has no particular behavior specific to a material. The dye is initialized in a discrete Gaussian blob at 800 m depth between 28 and 29° N latitude. Building off information from the previous simulations, only two offline simulations were run: one with the online output frequency forced at `nhis` = 100 (about 30 min) to be used as a "good-
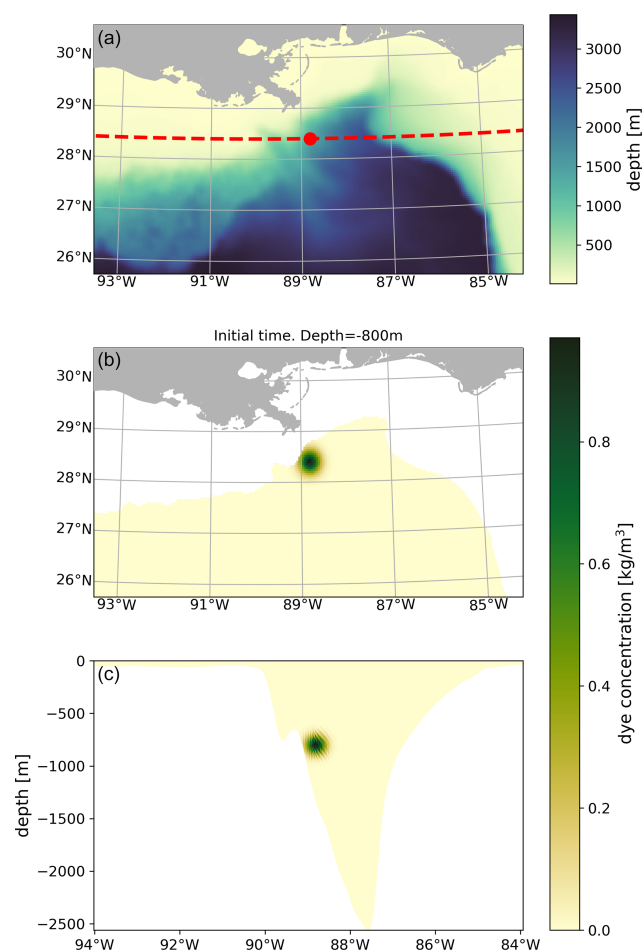

Figure 2. Second experiment: discrete Gaussian blob of dye at 800 m depth. Subplots show the domain and bathymetry **(a)**, the dye slice at 800 m depth **(b)**, and the vertical cross section of the dye field **(c)** across the red dashed line in panel **(a)**. The red circle in panel **(a)** indicates the center of the dye blob.

resolution" test case, and one with the online output frequency forced at `nhis` = 1000 (about 5.5 h) to be used as a "low-resolution" test case; both were run with the `U3C4` tracer advection scheme and an offline time step of 20 times the online time step, or 400 s.

2.3 Metrics

2.3.1 Skill score

The main metric used to evaluate the performance of this model is a skill score, SS (Bogden et al., 1996; Hill et al., 2004; Hetland, 2006). This is calculated as follows:

$$SS = 1 - \frac{\sqrt{\langle (D_{\mathrm{on}} - D_{\mathrm{off}})^2 \rangle}}{\sqrt{\langle D_{\mathrm{on}}^2 \rangle}}, \qquad (1)$$

where D_{off} and D_{on} are the volume of dye on the 3D grid and in time for the off- and online simulations, respectively,

and the brackets $\langle . \rangle$ indicate averaging over horizontal and vertical dimensions, returning a time series.

Often skill scores are calculated with respect to a reference. For example, for numerical model performance, the difference between model and data in the numerator may be compared with the difference between climatology and data in the denominator in order to assess how much better the model is performing than simple climatology (Hetland, 2006). An analogous comparison may be made here vs. persistence of the initial condition of the dye patch, so that this skill score shows how well the offline model performs compared with simply persisting the initial condition:

$$SS_p = 1 - \frac{\sqrt{\langle (D_{off} - D_{on})^2 \rangle}}{\sqrt{\langle (D_{initial} - D_{on})^2 \rangle}} \qquad (2)$$

Skill scores are a comparison between an offline simulation and the online simulation from which it is forced, unless otherwise noted; thus, the skill score represents the accuracy of the offline simulation to the online simulation, or the skill in faithfully reproducing the online simulation. This is different from a measure of the accuracy of the online simulation itself to simulate the dynamics.

2.3.2 Percent error

Percent error is used to demonstrate the accuracy of the second set of simulations in space, because it is not averaged over spatial dimensions like the skill score. The percent error at time t_0 is calculated as follows:

$$E(t_0) = \frac{|D_{on}(t_0, z, y, x) - D_{off}(t_0, z, y, x)|}{V_{on}(t_0, z, y, x) d_{max}(t_0)}, \qquad (3)$$

where $D_*(t_0, z, y, x)$ is the on- or offline dye volume at time t_0 in space (kg), $V_{on}(t_0, z, y, x)$ is the online volume of the grid cells (m^3), and $d_{max}(t_0)$ is the maximum dye concentration at t_0 (kg m^{-3}). The percent error represents the difference in the offline from the online simulation compared with the maximum possible dye mass at that time step.

2.4 Simulations and software

Simulations were performed on a Linux cluster with 84 processors for online simulations and 28 processors for offline simulations. The number used was not optimized. Analysis was performed in a Jupyter notebook (Kluyver et al., 2016) using pandas (McKinney, 2010), xarray (Hoyer and Hamman, 2017), and SciPy (Virtanen et al., 2020) for analysis, and Matplotlib (Hunter, 2007) for figures with cmocean (Thyng et al., 2016) for colormaps.

3 Results

3.1 Full water column Gaussian

The accuracy of selected offline simulations is presented here. As there were over 300 offline simulations, only selected results are shown to best illustrate specific points and show the overall performance of the model under a range of parameter choices. Offline simulations are forced by snapshots of online output (`his`, not `avg` files) in all cases unless specified. These results are specific to this model setup and the dynamics that are being captured in the region, but they should give specific results for other geographically interested users with similar model setups and general guiding results for others.

Instantaneous differences in dye concentration demonstrate the spatial structure of the offline simulation errors (Fig. 3). The structure changes not just with changes in the frequency of forcing in the offline simulation (`nhis`) and offline time step (`dt`) but also with the tracer advection scheme used. Comparing the top two rows in Fig. 3, we see that the error in the `MPDATA` simulations tends to be more localized when compared with the `U3C4` simulations. The magnitude of error increases with both a decrease in forcing frequency and an increase in time step for the offline simulations (moving from subplots A to C); in particular, the `MPDATA` simulation shows much more widespread spatial structure in the errors with `dt` = 20 (subplots C). Subplots E and F show fairly similar structure across the simulations, although with larger errors for `MPDATA`. Subplots D show the much larger errors that result when the vertical salinity diffusion coefficient `Aks` is not forced in the offline simulation.

Skill scores (Eq. 1) over time, demonstrating offline model accuracy, are shown in Fig. 4, and a summary is shown in Table A1. Both tracer advection schemes (`U3C4` and `MPDATA`) give highly accurate results (Fig. 4a), though `U3C4` performs a bit better than `MPDATA`. When vertical salinity diffusivity, `Aks`, which controls the impact of sub-grid-scale vertical mixing on the tracer field, is not forced (`Aksoff`), offline accuracy is reduced, though just 2 percentage points over 14 d compared with when it is forced. The impact of how often online model output is saved and input into the offline simulation, controlled by the `nhis` parameter, is almost negligible below 200 or 500 times the online time step (`nhis` 200, about 1.1 h, and `nhis` 500, about 2.8 h, respectively), but it has increasing impact for less frequent online model output (higher values of `nhis`, Fig. 4b). This means that for the present model setup and region, the frequency of online output higher than about 1–3 h is not important. Results are relatively similar with `nhis` 1000 (about 5 h), but accuracy decreases significantly as `nhis` increases beyond that. Context for the `nhis` values is given in Sect. 4.

The importance of `nhis` and the offline time step together for tracer advection scheme `MPDATA` is shown in Fig. 4c. The largest control on the skill score is from `nhis` – the values
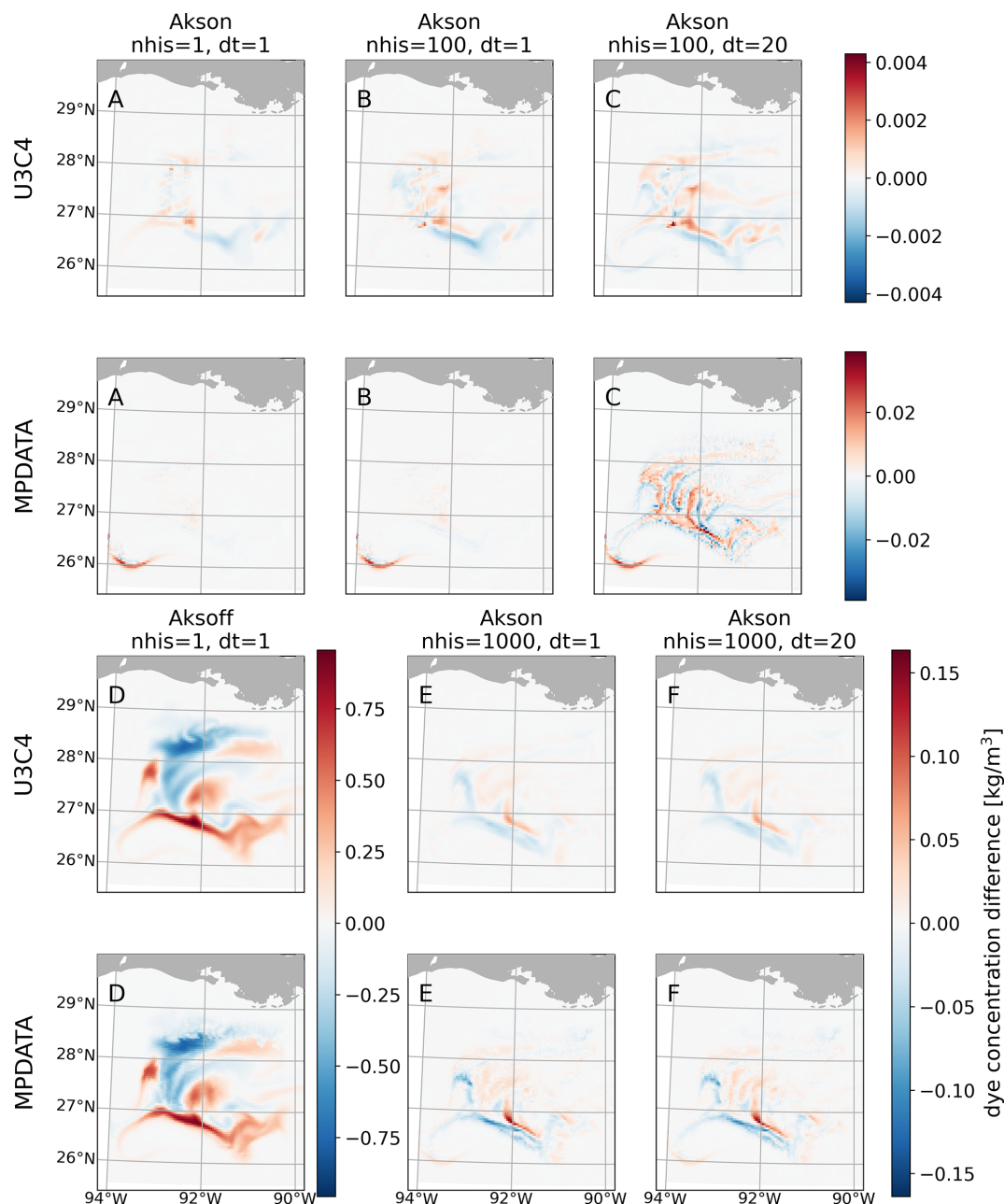
Figure 3. Instantaneous difference in dye concentration (online minus offline simulation) after about 13.2 d. Alternating rows show the results from the two tracer advection schemes tested, and the columns show the different experiments. All pairs of experiments except D forced the vertical salinity diffusion coefficient `Aks`. Experiment A vs. B shows the result of changing the forcing frequency of the online output into the offline simulation, `nhis`, from 1 (every online time step) to every 100 online time steps for the online time step. Experiment C shows the result of additionally changing the offline time step to be 20 times the online time step. Experiments E and F show offline experiments forced with online output every 1000 online time steps with the offline time step of the online time step (dt = 1) or 20 times the online (dt = 20) time step. Note that each color bar has a different range of values.

shown demonstrate the spread from the highest accuracy to several levels down (`nhis` 200, 500, and 1000 times the online time step, or about 1, 3, and 5.5 h, respectively). For each `nhis` value, three different offline time steps, `dt`, are shown (dt of 1, 10, and 20 times the online time step). Accuracy decreases with increasing offline time step but in different relative amounts that depend on the `nhis` value. For `nhis` of 200 and 500, there is more impact from the change in offline time step `dt` than from the `nhis` value. However, for `nhis` 1000, the `dt` values do not strongly impact the results.
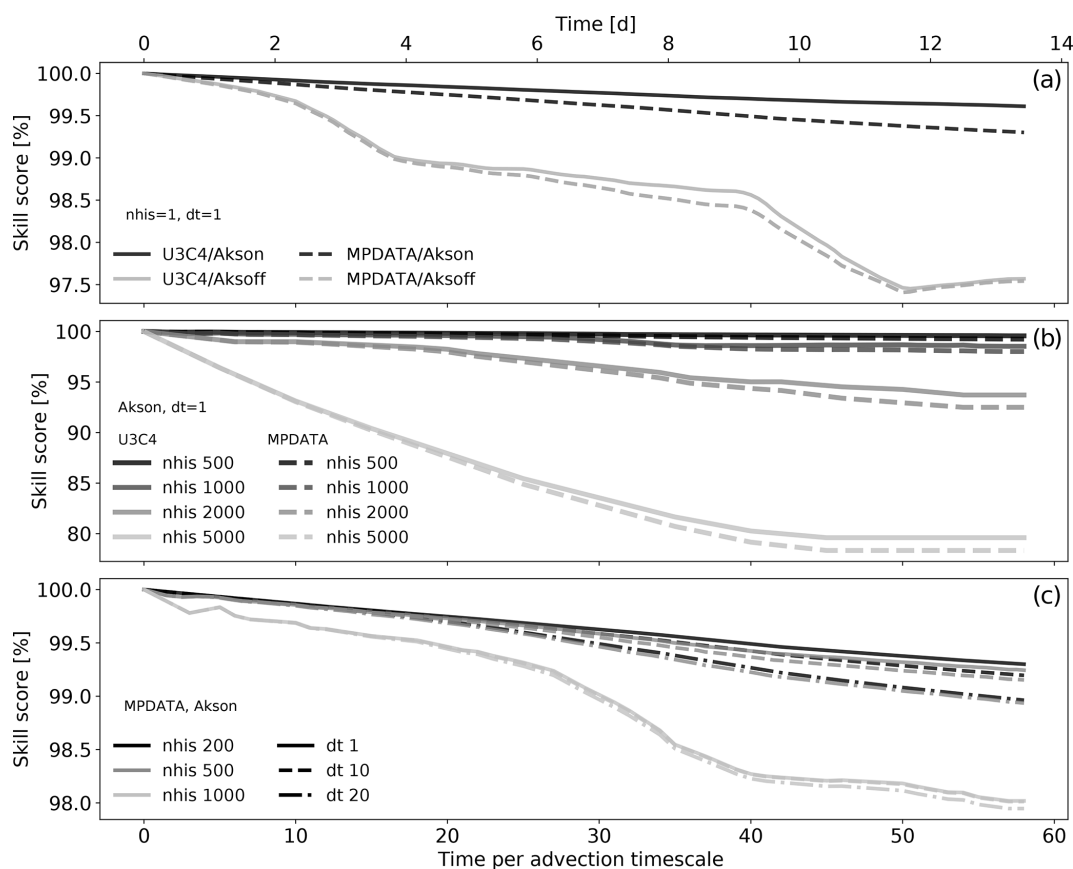
Figure 4. Skill scores for several subsets of offline simulations. **(a)** Performance between the `MPDATA` and `U3C4` tracer advection schemes and whether the vertical salinity diffusion coefficient `Aks` is forced with the online simulation ("on") or a constant background number ("off"). These cases also have `nhis`=1 (the online output was saved each time step) and `dt`=1 (the offline time step matched the online time step). **(b)** Performance between the `MPDATA` and `U3C4` advection schemes with `nhis` values varying. These cases also have `Aks` forced from the online case and `dt`=1. **(c)** Performance for varying `nhis` and `dt` parameters, where `MPDATA` is used and `Aks` is forced from the online case. All offline simulations here are forced by `his` files.

Offline time step results for `U3C4` simulations are not shown because the time step does not strongly impact results for any `nhis` values.

Several issues are demonstrated in Fig. 5. First is an example of model performance for a skill score based on persistence (Eq. 2). Model performance is similar, though a little lower, when assessed using the persistence skill score as compared with the regular skill score, so it is only shown here. This tells us that the offline model does indeed provide more benefit than simply persisting the initial condition. Next is a demonstration of offline accuracy compared to online output when different tracer advection schemes are used (Fig. 5b). For reference, simulations forced with the same tracer advection schemes both on- and offline are shown as well (`U3C4`, black solid, and `MPDATA`, black dashed). We find a significant decrease in offline model accuracy when the offline advection scheme does not match the online scheme, because different numerical schemes have different numerical dispersion and diffusion properties leading to differ-

ences in tracer advection. For comparison, the "skill score" comparing online `U3C4` and online `MPDATA` output (gray dashed) is shown. The online–online comparison for the two schemes has comparable performance, though lower; it is not clear if there is a reason that the on- and offline combinations should be better or worse than this, but the issue was not further explored. The best fidelity to an online simulation will be found by forcing the offline simulation with the same tracer advection scheme as that used in the online simulation. Also, forcing the offline simulation with a different tracer advection scheme from the online simulation will give results that are different from the online results on the order of the difference between the results of the different tracer advection schemes themselves. Finally, the significant impact of using single-precision output is demonstrated (Fig. 5c); it is best to save online model output for forcing offline simulations with double precision.

Several other issues were investigated but not plotted (they can be seen in the paper GitHub repository). Passive tracers
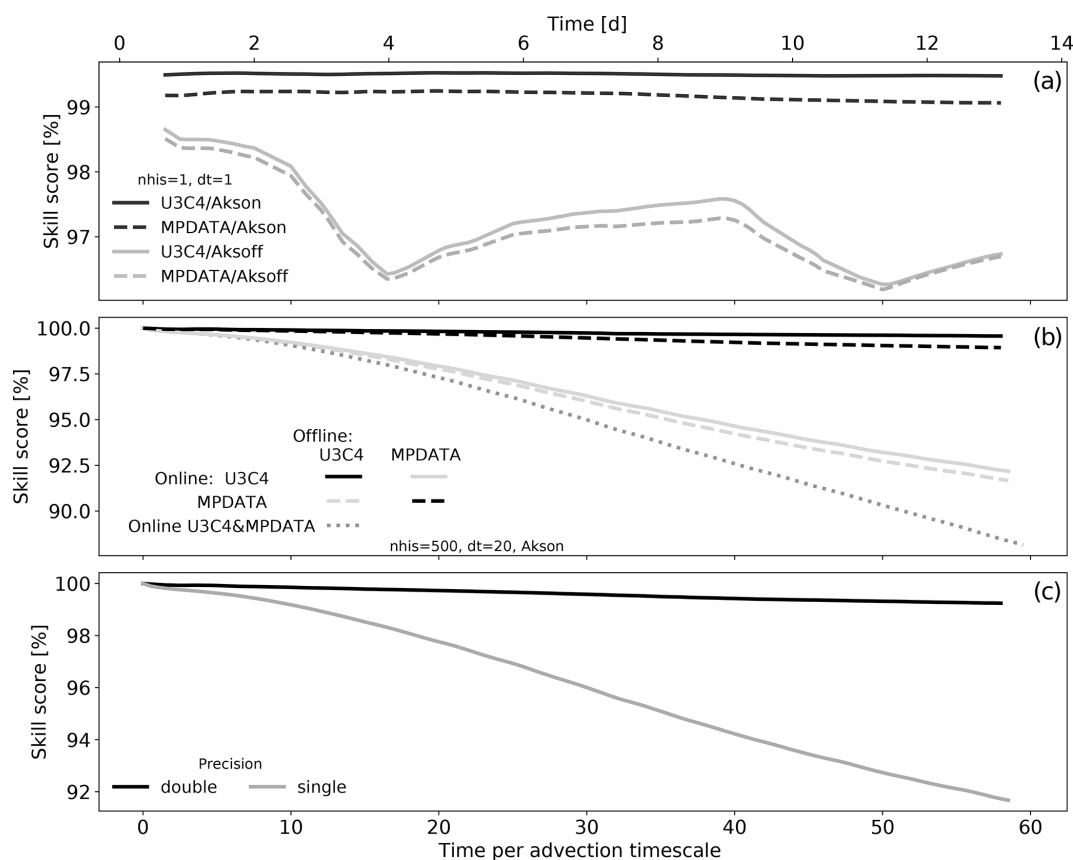
Figure 5. (a) Skill score compared with persistence is shown for combinations of tracer advection schemes and whether online `Aks` is forced. The simulations shown are the same as in Fig. 4a. **(b)** Comparison of skill score simulations forced by several combinations of tracer advection schemes. The combinations are as follows: online simulation using `U3C4` with offline simulation using `U3C4` (black solid line), online simulation using `U3C4` with offline simulation using `MPDATA` (gray solid), online `MPDATA` with offline `U3C4` (gray dashed) and with offline `MPDATA` (black dashed), and a comparison between results from online `U3C4` and online `MPDATA` (gray dotted). **(c)** Skill score for double-precision compared with single-precision online output.

are conserved in online ROMS simulations (Shchepetkin and McWilliams, 2005); offline simulations also conserve tracers. Only small differences were found between forcing offline simulations with snapshots (`his` files) or averages between time steps (`avg` files) from online simulations. Finally, for simulations in which a realistic `Aks` field was not forced, the background value used for `Aks` was varied; we found that this did not impact results.

An overview of results is shown in Fig. 6. The objective of this figure is to display the competing factors – computation time (x axis) and storage required (y axis) – that will ultimately determine offline accuracy (colored markers). Skill scores are shown for four subsets of simulations: tracer advection scheme `U3C4` with (diamonds) and without (downward-pointing triangles) `Aks` realistically forced, and tracer advection scheme `MPDATA` with (squares) and without (upward-pointing triangles) `Aks` realistically forced. The best compromise between storage, computational time, and skill score is where the skill score is still high – in one of the top classes, but with the lowest storage and time re-

quirements. For the present set of simulations, this occurs for `U3C4` with realistic `Aks` for `nhis` of 200 (more conservative) or 500 and `dt` of 20, and for `MPDATA` with realistic `Aks` for `nhis` 200 and `dt` of 5. Simulations in which `Aks` is not forced always have lower accuracy, and the small storage saving is probably not worth the loss; however, there may be circumstances in which online `Aks` is not available.

3.2 At-depth realistic Gaussian

The biggest difference in the second set of simulations (initialization shown in Fig. 2) compared with the first is the variation in the vertical direction: a dye blob was initialized at a particular depth instead of throughout the water column. A skill score comparison between 13 and 14 d indicates that the good-resolution experiment (`nhis` = 100, or online output forced in the offline simulation every ∼ 30 min) had about the same skill score of 99.6 % as the comparable previous numerical experiment skill score. However, the low-resolution test case (`nhis` = 1000, or online output forced
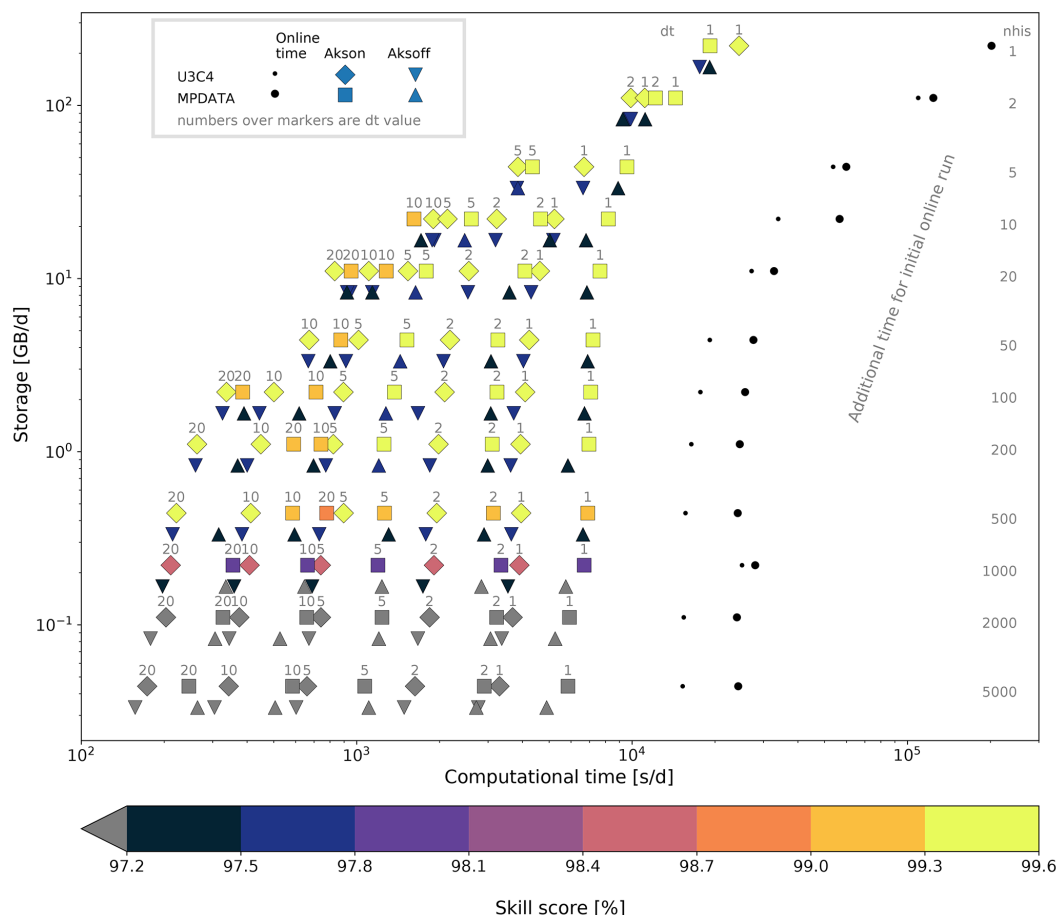
Figure 6. Summary of skill score results. Shown are the offline computational time per simulation day (*x* axis), the storage required for the online simulation per simulation day (*y* axis), and the skill score after about 13.5 d of simulation when forcing with snapshots (`his` files) for a range of `nhis` and `dt` values (colored markers, with one set for forcing `Aks` or not, and which tracer advection scheme is used). `nhis` values for rows are indicated on the right-hand side of the plot, and `dt` values are shown above each pair of markers. Values below 97.2 % are colored gray. The computational time required for the online simulation is shown separately with black markers.

every ∼ 5.5 h) had a much lower skill score of 70 % compared with the first test case of 98.5 %, possibly indicating a compensatory effect in the first set of experiments in the vertical direction. That is, dye may have been transported vertically inaccurately in the first set of lower-resolution experiments, but as the whole water column had dye in it, it may have still given better skill scores than if the dye patch was instead discrete.

Spatial differences in the accuracy of the experiments are shown for depth slices (Fig. 7) and cross sections (Fig. 8). The dye in the good-resolution cases stays close to the online simulation, with small differences in the percent error near where the dye encounters the bathymetry on the west end of the blob (Fig. 7d and f, noting that the values in panel d have been multiplied by 100 to be visible). The low-resolution case is qualitatively similar to the online case, but the difference (Fig. 7e) shows patches of large disagreement. The disagreement is further demonstrated in the low-resolution

case percent error (Fig. 7g) with a swath of 1 %–10 % error across the full dye feature.

Results are similar for the vertical cross section (Fig. 8). The differences in the offline and online dye field are very small in the good-resolution case – it has been multiplied by 500 to appear on the same color bar as the low-resolution case. In the low-resolution case, the offline dye has been transported both up and down more than in the online case.

4   Discussion

The context of the performance difference found as a function of `nhis` values (Fig. 4b, c) can be considered as the impact of loss of energy represented in the system (an approach also used by Qu and Hetland, 2019). For example, Fig. 9 shows the power spectral density of the online simulation speed from near the middle of the dye patch. The output frequency, `nhis`, from the online simulation controls how
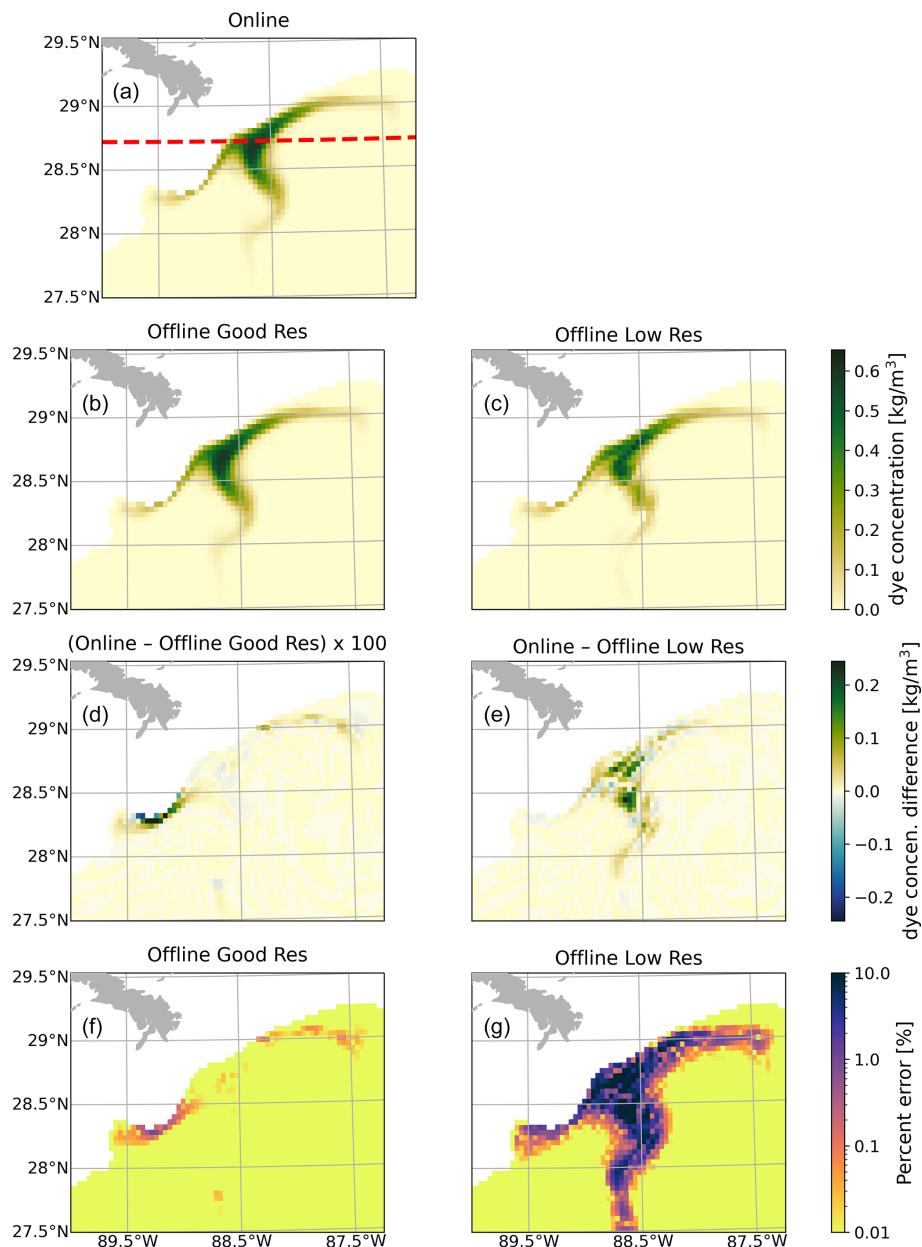
Figure 7. Snapshots of the dye at 13.75 d for the online (**a**), good-resolution offline (**b**), and low-resolution offline (**c**) simulations. The difference in dye concentration for the online and offline cases at the same time for the good-resolution (**d**) and low-resolution (**e**) experiments. The percent error for the good-resolution (**f**) and low-resolution (**g**) offline cases is also shown. Moreover, panel (**a**) indicates the slice location shown in Fig. 8. Note that values in panel (**d**) have been multiplied by 100 to be visible on the same color bar as panel (**e**) as the differences are so small.

much energy of this spectrum the offline simulations receive and, therefore, how much of the system's energy is represented offline. The amount of energy missing can be seen visually by the overlaid lines representing different output frequencies, nhis. Skill score results (Fig. 4) show that accuracy decreases as nhis values increase starting at nhis of 200 or 500 (about 1 to 3 h), which correspond to between

1 % and 5 % of the total energy being lost to subsampling the output.

Comparing a relevant dynamical timescale to nhis is another way to provide context for its impact on offline accuracy. A previous study evaluating an offline tracer from MIT-gcm model output found that for their global-scale model, the inertial period controlled the output rate necessary for robust results (Hill et al., 2004). We find an analogous result here,
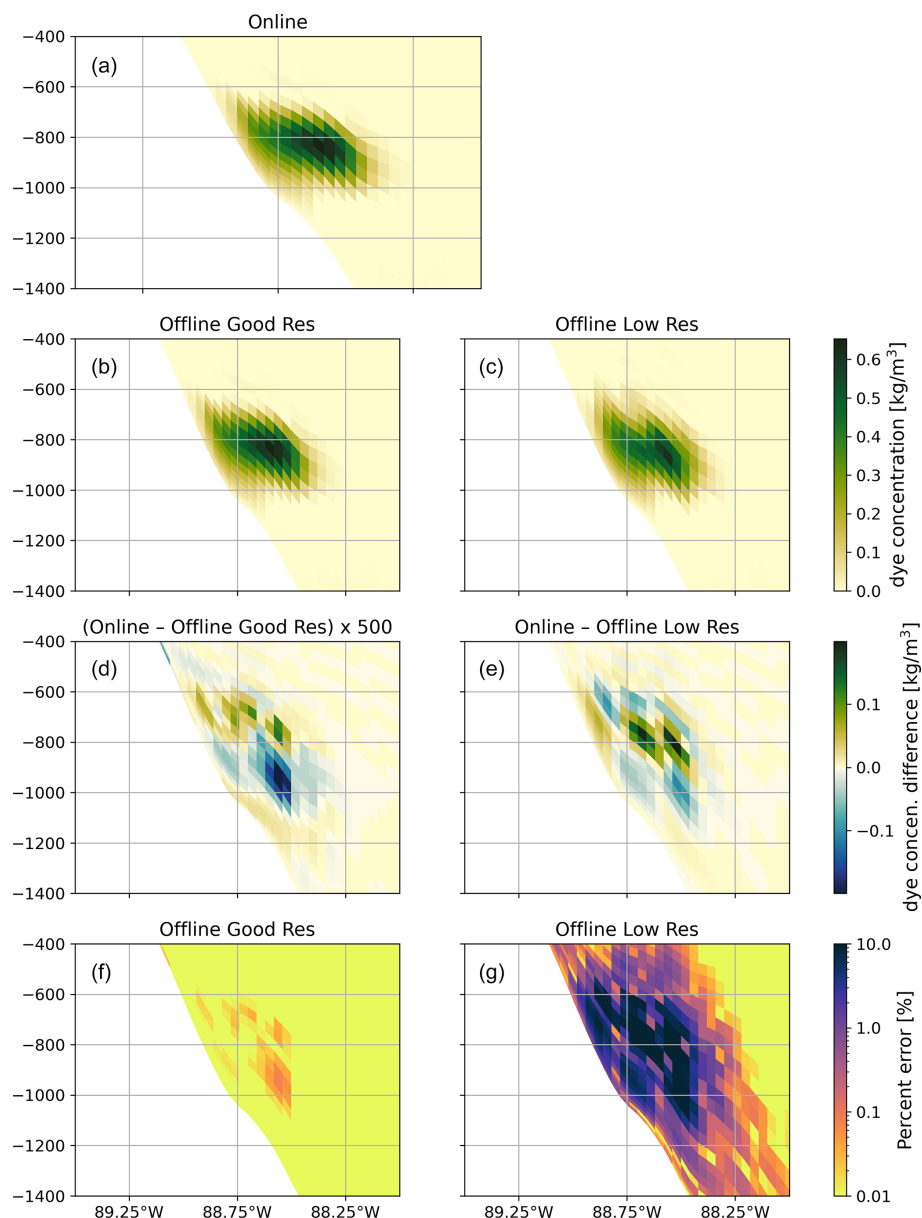
Figure 8. Vertical cross section comparisons of the online and offline simulations; the cross section location is indicated in Fig. 7a. Snapshots at 13.75 d are shown for the online **(a)** and offline good-resolution **(b)** and low-resolution **(c)** cases. Differences at the same time are shown in panels **(d)** and **(e)**. The percent error is shown in panels **(f)** and **(g)**. Note that values in panel **(d)** have been multiplied by 500 to be visible on the same color bar as panel **(e)** as the differences are so small.

though the relevant timescale is the advection timescale (see Sect. 2.1). The advection timescale for this regional model is about 20 000 s, which corresponds to an output rate from the online model of `nhis` = 1000 times the online time step, which is indeed the turning point for clear degradation in offline model accuracy we find (Fig. 6).

We should expect that the offline time step is controlled by the horizontal Courant number and that our results destabilize as the number increases toward 1. An estimate of the horizontal Courant number, with the largest horizontal ve-

locity of 1 and the smallest horizontal cell width of about 3800 m, for the offline time steps gives a range from 0.005 for the offline time step matching the online time step up to about 0.1 and 0.25 for offline time step `dt` of 20 and 50, respectively. Simulations gave reasonable results for `dt` of 20 but not for `dt` of 50.
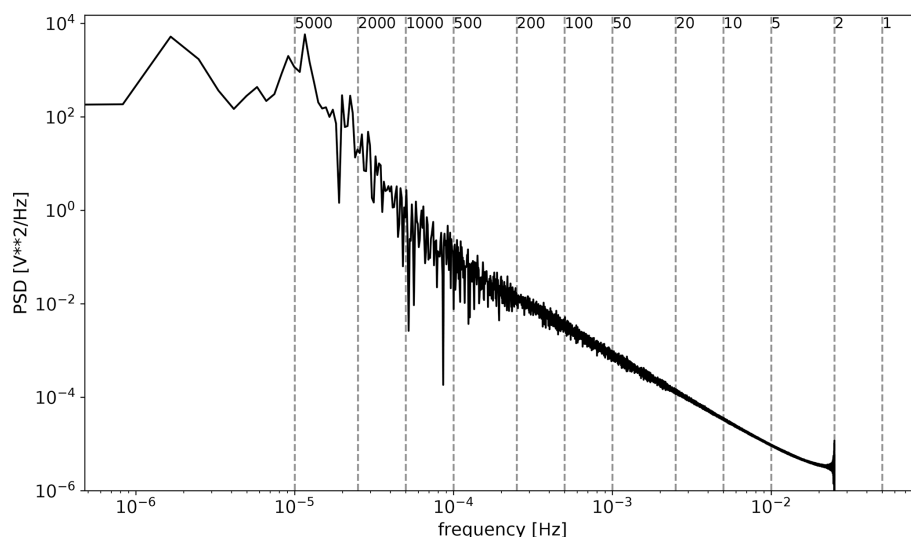
Figure 9. Power spectral density for speed at a single location near the center of the dye patch. Overlaid (gray dashed) are lines marking frequencies at which online model output was saved for forcing offline simulations; these are marked with their corresponding `nhis` value.

5 Conclusions

This paper presents a description and evaluation of an offline tracer advection model developed within ROMS. The advantage of this is the ease and consistency with which ROMS users can employ existing model output to force offline tracer simulations at low computational cost. The main approach of the offline model is to force variables `zeta`, `u/v`, and `ubar`/`vbar` from an online simulation as climatology; normally climatology would be used in a ROMS simulation to nudge boundary conditions toward mean values, but in this case all grid cells are fully forced. Additionally forcing the vertical salinity diffusivity, `Aks`, improves model accuracy. It is also important that the online simulation output used to force the offline simulation has double precision.

We tested two tracer advection schemes, `MPDATA` and `TS_U3HADVECTION` with `TS_C4VADVECTION` (third-order upstream horizontal advection and fourth-order centered vertical advection, called `U3C4` here), in a regional simulation of the northern Gulf of Mexico, and we found that the offline simulations are able to reproduce online simulations at a high accuracy. The most important control differentiating offline accuracy was the `nhis` parameter describing how often online simulation output was saved, as a multiple of the online time step, to be input into the offline simulation. For both tracer advection schemes and with `Aks` forced, the offline simulations showed high accuracy up to `nhis` = 200 or 500, about 1.1 and 2.8 h, respectively. This is consistent with requiring temporal information at a rate higher than the relevant dynamic timescale – in this case, an advection timescale approximated as roughly equivalent to an `nhis` value of 1000. The offline time step `dt` was not an important choice for offline simulations run with `U3C4`, as long as it was under about 50 (all had skill scores of 99.6 % after

14 d). However, for `MPDATA` offline simulations were highly accurate with a time step 5 times the online time step up to `nhis` = 200, with some dependence on the offline time step.

A second set of simulations were run to demonstrate performance in a more realistic, application-driven experiment – in this case, with a discrete blob of dye at depth. The good-resolution case with online forcing at a frequency of `nhis` = 100 (about 30 min) was very accurate, with a similar skill score to the original comparable offline `U3C4` experiment run of 99.6 %. The low-resolution experiment of `nhis` = 1000 (about 5.5 h) gave worse results than the comparable previous simulation, implying that the vertical direction is indeed important and can behave distinctly from the horizontal. Overall, the results show that it is possible to get high-fidelity results in offline tracer simulation with this code.

Appendix A: Table of skill scores

Table A1. Final skill score (percent) of offline simulations after 14 d, sorted by `nhis` and `dt` values, tracer advection scheme, and if `Aks` is forced.

	Advect	MPDATA		U3C4	
	Aks	Off	On	Off	On
nhis	dt				
1	1	97.5	99.3	97.6	99.6
2	1	97.5	99.3	97.6	99.6
	2	97.5	99.3	97.6	99.6
5	1	97.5	99.3	97.6	99.6
	5	97.6	99.3	97.6	99.6
10	1	97.5	99.3	97.6	99.6
	2	97.5	99.3	97.6	99.6
	5	97.6	99.3	97.6	99.6
	10	97.5	99.2	97.6	99.6
20	1	97.5	99.3	97.6	99.6
	2	97.5	99.3	97.6	99.6
	5	97.6	99.3	97.6	99.6
	10	97.5	99.2	97.6	99.6
	20	97.5	99.0	97.6	99.6
50	1	97.5	99.3	97.6	99.6
	2	97.5	99.3	97.6	99.6
	5	97.6	99.3	97.6	99.6
	10	97.5	99.2	97.6	99.6
100	1	97.5	99.3	97.6	99.6
	2	97.5	99.3	97.6	99.6
	5	97.6	99.3	97.6	99.6
	10	97.5	99.2	97.6	99.6
	20	97.5	99.0	97.6	99.6
200	1	97.5	99.3	97.6	99.6
	2	97.5	99.3	97.6	99.6
	5	97.6	99.3	97.6	99.6
	10	97.5	99.2	97.6	99.6
	20	97.5	99.0	97.6	99.6
500	1	97.5	99.2	97.6	99.6
	2	97.5	99.2	97.6	99.6
	5	97.5	99.2	97.6	99.6
	10	97.5	99.2	97.6	99.6
	20	97.5	98.9	97.6	99.6
1000	1	96.9	98.0	97.2	98.5
	2	96.9	98.0	97.2	98.5
	5	96.9	98.0	97.2	98.5
	10	96.9	98.0	97.2	98.5
	20	96.9	97.9	97.2	98.5
2000	1	92.2	92.5	93.4	93.7
	2	92.2	92.5	93.4	93.7
	5	92.2	92.5	93.4	93.7
	10	92.2	92.5	93.4	93.7
	20	92.3	92.6	93.4	93.7
5000	1	78.2	78.3	79.4	79.6
	2	78.2	78.3	79.4	79.6
	5	78.2	78.4	79.4	79.6
	10	78.2	78.4	79.4	79.6
	20	78.3	78.4	79.4	79.6

Appendix B: Explanation of code changes

While preprocessor flags for offline simulations already existed in the ROMS and COAWST code base, we found that the offline simulations did not work as desired. In this section, we describe changes made to the code base so that offline passive tracer advection works properly by receiving the necessary forced variables. Generally, the offline code works by forcing previously simulated online model output that is input as climatological forcing. Typically, climatology would be used in a ROMS simulation to nudge boundary conditions toward mean values, but in this case all grid cells are fully forced.

Code changes were made to avoid repeating processes offline that were already included online. Initialization is now minimal for offline simulations (`ini_fields.F`), and initial values are replaced by the first time step read in from climatology. Updates to sea surface height `zeta` (calls to `ini_zeta` and `set_zeta` in `main3d_offline.F`) have been removed as the variable is directly forced in the offline simulation. Boundaries are not forced in the offline case (except for the passive dye field): horizontal indices now start one index earlier and end one index later in each tile so that climatology is read into ghost cells in place of boundary conditions (`set_data.F`). The remaining processes are controlled through the user input file and preprocessor flags (Appendix C).

The offline simulation is missing much of the complex time stepping in an online ROMS simulation due to the missing numerics, leading to necessary code adjustments (`set_data.F`). Climatology for 3D variables (`u`/`v`, `salt`/`temp`, and `tke`/`gls`) are read into earlier time indices (`nrhs` instead of `nnew`) to account for this, eliminating a time shift that otherwise occurs. Model output for the subsequent time step are read in from climatology and saved in available time indices for several variables (`zeta`, `Aks`, and `Akt`) to be used later in the time loop. In the online simulation, `zeta` is normally updated mid-time loop with the fast time-stepping value. To approximate this behavior, the two time steps of `zeta` are averaged into variable `Zt_avg1` (new function `set_avg_zeta`). Calculations of vertical layer thickness `Hz` and mass fluxes `Huon`/`Hvom` for the subsequent time step are made mid-time loop in `set_depth.F`. New functions `set_massflux_avg` and `set_massflux_avg_tile` were added to `set_massflux.F` to average `Huon`/`Hvom` values to be used subsequently in `step3d_t.F` where the tracer is advected. This change matches the online case to floating-point round-off error.

The OFFLINE preprocessor flag with OFFLINE_TPASSIVE compiles the necessary code to run offline tracer advection (more details in Appendix C). These flags already existed, but changes to the code for the present project were made under these flags. Other available offline preprocessor flags include

- `omega` (OCLIMATOLOGY), which already existed and does not impact offline tracer advection, reads in climatology for the S coordinate vertical momentum component.

- `Aks` (AKSCLIMATOLOGY), `Akt` (AKTCLIMATOLOGY), and `Akv` (AKVCLIMATOLOGY), or all three `Aks`, `Akt`, and `Akv` (AKXCLIMATOLOGY) are new flags. `Aks`, the vertical salinity diffusion, impacts the accuracy of the offline tracer (Sect. 3), whereas `Akt`, the vertical temperature diffusion, does not impact offline tracer advection; the latter is used for offline floats (OFFLINE_FLOATS) if vertical walk (FLOATS_VWALK) is activated. `Akv`, the vertical viscosity, does not impact offline passive tracer advection.

- `TKE` (TKECLIMATOLOGY, turbulent kinetic energy), `GLS` (GLSCLIMATOLOGY, generic length scale), or both (MIXCLIMATOLOGY) are new flags. These do not impact offline passive tracer advection.

- `salt` and `temp` (ATCLIMATOLOGY) are new flags. These flags are also impacted by `LtracerCLM` in the input file. While these do not impact offline tracer advection, they may be used for other modules such as oil modeling with offline floats.

To fix a problem with reading in the climatology at the correct time step, a condition was added (`get_2dfld.F` and `get_3dfld.F`) that compares the differences in times to being less than half a time step, avoiding any problems with numerical precision.

`omega`, the mass flux perpendicular to the local S coordinate, was already set up to be read in through climatology with the OCLIMATOLOGY flag, but results did not match on- and offline. The lower vertical index in the call for omega in `get_data.F` was one, which is used for `rho` grids instead of the w-grid `omega` is actually on, which starts at index zero.

Appendix C: How to set up simulations

Requirements and considerations for setting up online and offline simulations in ROMS or COAWST with the offline passive tracer advection code are provided below.

C1 Online

C1.1 Input file

In the project input file (the `*.in` file, for example, https://github.com/kthyng/oil_03/blob/master/External/ocean_oil_03.in, last access: 18 May 2020), the items below should be considered in addition to the typical input parameter selections:

- Choose whether to save output as snapshots at a single time or averages across time intervals (ROMS `his` vs. `avg` files). Your choice will be used to force the offline simulation. Present results show that this choice does not significantly change results. We recommend using `his` files in the absence of any other preference as if `his` files are not used, it is necessary to include the initial file prepended to the input `avg` file in `CLMNAME`.

- Output necessary variables for forcing the offline simulation. Variables `zeta`, `ubar`, `vbar`, `u`, and `v` are required for forcing the offline simulation, and `Aks` is optional for improved accuracy in the offline simulation (though it increases amount of storage required).

- Choose output frequency (parameter `NHIS` for `his` files or `NAVG` for `avg` files). This is how often ROMS will save output to a `his` or `avg` file, as a multiple of the time step, and in turn this is what will be used to force the offline simulation. Important considerations for this selection include acceptable simulation runtime and storage requirements. Figure 6 gives a paradigm from which to decide this for simulations in general. In the present study, for `U3C4` there was a drop in performance below an output frequency of 500 times the online time step, and below 200 or 500 times for `MPDATA`. These choices will vary for a given model setup and accuracy needs.

- For this online simulation, point to file `varinfo-online.dat` for `VARNAME`, which is a typical, unchanged file. This has been provided in the code repository: https://github.com/kthyng/COAWST-ROMS-OIL/blob/master/ROMS/External/varinfo-online.dat (last access: 18 May 2020).

C1.2 Header file

In the project header file (the `*.h` file, for example, https://github.com/kthyng/oil_03/blob/master/Include/oil_03.h, last access: 18 May 2020), the following additional flags should be considered:

- Choose a tracer advection scheme. We tested two schemes and found both accurately reproduced the online results offline, though `U3C4` performed slightly better. Note, however, that online tracer advection performance itself depends on the dynamics involved; more information is available in Kalra et al. (2019). Also note that `MPDATA` requires more runtime than `U3C4` (Fig. 6).

- Use `OUT_DOUBLE` to output results with double precision to significantly improve your accuracy, though increase storage is required (Fig. 5).

C2 Offline

C2.1 Input file

In the project input file (the `*.in` file, for example, https://github.com/kthyng/oil_off/blob/master/External/ocean_oil_offline.in, last access: 18 May 2020), the items below should be considered in addition to the typical input parameter selections:

- The output frequency (`NHIS` or `NAVG`) will not impact your offline simulation performance, but it should be chosen to well represent the dynamics in your model.

- A reasonable choice for the offline simulation time step `dt` is a multiple of the online time step. Some testing for your model setup is warranted. The present study found that time step was not important for the `U3C4` tracer scheme combination – a `dt` of 20 times the online time step gave accuracy which was as good as that for the online time step itself. However, for `MPDATA`, only using the online time step gave the highest accuracy; for the next level down of accuracy, a time step of 10 times the online time step was adequate (Fig. 6). Note also that the offline time step needs to factor evenly into the online output frequency, and the offline time step cannot be larger than the online output frequency.

- All physics should be off in the offline case, except for anything directly impacting the offline tracer field (`dye_01`) itself, because it is included in the online output. This implies that

  - boundaries should all be closed except for offline tracer fields, e.g., parameter `LBC(isFsur)`;
  - river forcing and other sources or sinks that were forced in the online simulation should be turned off;
  - winds, bulk fluxes, etc, should not be forced from the online simulation;
  - the model should not be nudged to climatology, even if used in the online simulation (climatology, the output from the online simulation, will be entirely enforced);
  - flags for climatology forcing for sea surface height (`LsshCLM`) as well as 2D (`Lm2CLM`) and 3D momentum (`Lm3CLM`) should be turned on, and salt and temperature flags (`LtracerCLM`) should also be turned on if you want to read them in (see Appendix C2.2).

- Only the sea surface height (zeta) and the offline dye(s) (`dye_01`) need to be output – other fields are best used directly from the online simulation (the vertical velocity w, for example, is not calculated properly in the offline simulation). The sea surface height is necessary to properly calculate tracer advection fluxes.

- Input as the climatology forcing (`CLMNAME`) the online model output. If forcing with an `avg` file from the online simulation, it is necessary to place a file containing the initial conditions first; this is possible by inputting a list of file names.

- For this offline simulation, point to file `varinfo-offline.dat` for `VARNAME`, which has been edited to include the new variables that can be input as climatology and so that all climatology time variables are named `ocean_time`. The latter change allows for the online output to be input directly offline as climatology without processing the file to rename variable attributes. The file has been provided in the code repository: https://github.com/kthyng/COAWST-ROMS-OIL/blob/master/ROMS/External/varinfo-offline.dat (last access: 18 May 2020).

C2.2   Header file

In the project header file (the `*.h` file, for example, https://github.com/kthyng/oil_off/blob/master/Include/oil_offline.h, last access: 18 May 2020), the following additional flags should be considered:

- Use the `OFFLINE` flag for any offline simulation and, additionally, the `OFFLINE_TPASSIVE` flag for offline tracer advection.

- For the best results, use the same tracer advection scheme as the online run. The schemes do not have to match, but the skill score between the simulations will diminish substantially (Fig. 5) as they do not use the same numerics. We did not test other tracer advection schemes, but we have no reason to think they will not work offline.

- Forcing the vertical salinity diffusivity `Aks` as predicted by the online simulation gives better offline accuracy than not forcing it, though it requires storing the information from the online case. This can be forced with the `AKSCLIMATOLOGY` flag. More information on the offline flags is available in Appendix B.

References

Azevedo Correia de Souza, J. M., Powell, B., Castillo-Trujillo, A. C., and Flament, P.: The vorticity balance of the ocean surface in Hawaii from a regional reanalysis, J. Phys. Oceanogr., 45, 424–440, 2015.

Bogden, P. S., Malanotte-Rizzoli, P., and Signell, R.: Open-ocean boundary conditions from interior data: Local and remote forcing of Massachusetts Bay, J. Geophys. Res.-Oceans, 101, 6487–6500, 1996.

Chassignet, E. P., Hurlburt, H. E., Smedstad, O. M., Halliwell, G. R., Hogan, P. J., Wallcraft, A. J., Baraille, R., and Bleck, R.: The HYCOM (hybrid coordinate ocean model) data assimilative system, J. Marine Syst., 65, 60–83, 2007.

Cummings, J. A.: Operational multivariate ocean data assimilation, Q. J. Roy. Meteorol. Soc., 131, 3583–3604, 2005.

Cummings, J. A. and Smedstad, O. M.: Variational data assimilation for the global ocean, in: Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications (Vol. II), pp. 303–343, Springer, Berlin, Heidelberg, 303–343, 2013.

Dutkiewicz, S., Follows, M., Marshall, J., and Gregg, W. W.: Interannual variability of phytoplankton abundances in the North Atlantic, Deep-Sea Res. Pt. II, 48, 2323–2344, 2001.

Fairall, C. W., Bradley, E. F., Hare, J., Grachev, A. A., and Edson, J. B.: Bulk parameterization of air–sea fluxes: Updates and verification for the COARE algorithm, J. Climate, 16, 571–591, 2003.

Fox, D., Teague, W., Barron, C., Carnes, M., and Lee, C.: The modular ocean data assimilation system (MODAS), J. Atmos. Ocean. Tech., 19, 240–252, 2002.

Gillibrand, P. A. and Herzfeld, M.: A mass-conserving advection scheme for offline simulation of scalar transport in coastal ocean models, Ocean Model., 101, 1–16, 2016.

Hetland, R. D.: Event-driven model skill assessment, Ocean Model., 11, 214–223, 2006.

Hill, H., Hill, C., Follows, M., and Dutkiewicz, S.: Is there a computational advantage to offline tracer modeling at very high resolution, Researchgate, Geophys. Res. Abstr., 6, 2004.

Hoyer, S. and Hamman, J.: xarray: N-D labeled Arrays and Datasets in Python, J. Open Rese. Softw., 5, 10, https://doi.org/10.5334/jors.148, 2017.

Hunter, J. D.: Matplotlib: A 2D graphics environment, Comput. Sci. Eng., 9, 90–95, 2007.

Kalra, T. S., Li, X., Warner, J. C., Geyer, W. R., and Wu, H.: Comparison of Physical to Numerical Mixing with Different Tracer Advection Schemes in Estuarine Environments, J. Marine Sci. Eng., 7, 338, https://doi.org/10.3390/jmse7100338, 2019.

Khatiwala, S., Visbeck, M., and Cane, M. A.: Accelerated simulation of passive tracers in ocean circulation models, Ocean Model., 9, 51–69, 2005.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J. B., Grout, J., Corlay, S., and Ivanov, P.: Jupyter Notebooks-a publishing format for reproducible computational workflows, ELPUB, 87–90, 2016.

Lévy, M., Resplandy, L., Klein, P., Capet, X., Iovino, D., and Éthé, C.: Grid degradation of submesoscale resolving ocean models: Benefits for offline passive tracer transport, Ocean Model., 48, 1–9, 2012.

McKinley, G. A., Follows, M. J., and Marshall, J.: Mechanisms of air-sea CO$_2$ flux variability in the equatorial Pacific and the North Atlantic, Global Biogeochem. Cycles, 18, 2004.

McKinney, W.: Data Structures for Statistical Computing in Python, in: Proceedings of the 9th Python in Science Conference, edited by: van der Walt, S. and Millman, J., pp. 56–61, https://doi.org/10.25080/Majora-92bf1922-00a, 2010.

Qu, L. and Hetland, R. D.: Temporal resolution of wind forcing required for river plume simulations, J. Geophys. Res.-Oceans, 124, 1459–1473, 2019.

Saha, S., Moorthi, S., Pan, H. L., Wu, X., Wang, J., Nadiga, S., Tripp, P., Kistler, R., Woollen, J., Behringer, D., and Liu, H.: The NCEP climate forecast system reanalysis, B. Am. Meteorol. Soc., 91, 1015–1058, 2010.

Shchepetkin, A. F. and McWilliams, J. C.: The regional oceanic modeling system (ROMS): a split-explicit, free-surface, topography-following-coordinate oceanic model, Ocean Model., 9, 347–404, 2005.

Thyng, K.: kthyng/offline_analysis: Revised run files for offline passive tracer simulation in ROMS (Version v1.1), Zenodo, https://doi.org/10.5281/zenodo.4278115, 2020a.

Thyng, K.: Run files for online simulation (Version v1.1), Zenodo, https://doi.org/10.5281/zenodo.4455715, 2020b.

Thyng, K.: Output from running offline passive tracer in ROMS ocean model, Gulf of Mexico Research Initiative Information and Data Cooperative (GRIIDC), Harte Research Institute, Texas A&M University – Corpus Christi, https://doi.org/10.7266/7R0N3FX4, 2020c.

Thyng, K. and Ruiz Xomchuk, V.: Run files for offline passive tracer simulation in ROMS (Version v1.1), Zenodo, https://doi.org/10.5281/zenodo.4455760, 2021.

Thyng, K., Chen, X., and Morey, S.: ROMS input files, figshare, Collection, https://doi.org/10.6084/m9.figshare.c.5097350.v1, 2020a.

Thyng, K., Chen, X., and Morey, S.: ROMS Input Files (second source), Gulf of Mexico Research Initiative Information and Data Cooperative (GRIIDC), Harte Research Institute, Texas A&M University – Corpus Christi, https://doi.org/10.7266/YF0QPBFC, 2020b.

Thyng, K., Kobashi, D., Ruiz Xomchuk, V., Qu, L., Chen, X., and Hetland, R. D.: Offline passive tracer in ROMS (Version v1.1), Zenodo, https://doi.org/10.5281/zenodo.4455738, 2021.

Thyng, K. M., Greene, C. A., Hetland, R. D., Zimmerle, H. M., and DiMarco, S. F.: True Colors of Oceanography: Guidelines for Effective and Accurate Colormap Selection, Oceanography, 29, 9–13, 2016.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Jarrod Millman, K., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E. W., Vand erPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods, 17, 261–272, https://doi.org/10.1038/s41592-019-0686-2, 2020.

Warner, J. C., Armstrong, B., He, R., and Zambon, J. B.: Development of a coupled ocean–atmosphere–wave–sediment transport (COAWST) modeling system, Ocean Model., 35, 230–244, 2010.

Zhang, Y., Jaeglé, L., and Thompson, L.: Natural biogeochemical cycle of mercury in a global three-dimensional ocean tracer model, Global Biogeochem. Cycles, 28, 553–570, 2014.