World Ocean Circulation

Algorithm Theoretical Basis Document for Lagrangian drift of *Sargassum* and oil spills (Theme 3)

customer	ESA/ESRIN
ESA contract	ESA Contract No. 4000130730/20/I-NB
document reference	WOC-ESA-ODL-NR-009_ATBD_T3_Lagrangian_Drift_V3
Version/Rev	3.0
Date of issue	03/05/2022

Distribution List

	Name	Organization	Nb. copies
Sent to :	M.H. rio	ESA/ESRIN	ESA-STAR
Internal copy :	Project Manager	OceanDatalab	1 (digital copy)

Document evolution sheet

Ed.	Rev.	Date	Purpose evolution	Comments
1	0	26/05/2021	Creation of document	
2		03/05/2022	Update of document with v2 of product	

Contents

List Of Images

1. Introduction

1.1. Products summary

The classical L4 combined ocean state [T,S,U,V,W] products are complemented with added value products. These added value products consist of the simulation of the Lagrangian drift of two types of materials: oil and *Sargassum*. The algorithms to run these simulations are based on the open-source OceanParcels.org framework, which has been developed by Utrecht University (Netherlands) to simulate the dispersion of plastic material in the ocean. The simulations describe the trajectory of a particle (in this case *Sargassum* or oil) advected by a velocity field.

1.2. Scope & Objectives

This document is the Algorithm Theoretical Basis Document (ATBD) which is dedicated to the description and justification of the algorithms used in the generation of the lagrangian drifts of two types of particles: *Sargassum* and oil. The general objective is to use the OceanParcels framework to simulate the trajectories of *Sargassum* and oil spills for specific case studies. The existing software will be run and adapting to each type of particle behaviour and scenario:

- Lagrangian drift of *Sargassum*

The Lagrangian drift of *Sargassum* is simulated in the Tropical Atlantic region for the years 2018 and 2019. During these years, a high abundance of *Sargassum* was observed in this region. For instance, in June 2018 a monthly mean *Sargassum* of more than 20 million tons was observed (Wang *et al*., 2019). Actually, the year 2018 was one in which a high quantity of *Sargassum* was observed in the tropical Atlantic, and several studies have been conducted (e.g. Miron *et al*., 2020). The monitoring of *Sargassum*, in particular in the Caribbean Sea, is important as it has become a priority problem after the massive stranding in 2018 and its impact on the local ecosystems and tourism. The objective here is to simulate virtual trajectories from the locations of the data available during these two years. For the first case we use the drifters released in the field experiment by Miron *et al*. (2020) and for the second one, the locations of *Sargassum* meshes identified in satellite images by MeteoFrance. We try to find out how accurately we can represent the trajectories in the Tropical Atlantic (including the Caribbean Sea). The trajectories are calculated at 15m depth (only depth at which the input velocity data is available).

- Lagrangian drift of oil spills

The Lagrangian drift of oil spills is simulated in the North Atlantic region for the year 2011 and in the Kuroshio region in 2018. Oil spills in the northern part of the North Atlantic are very difficult to manage, hence rapid-response tools are particularly important here (Bobra and Fingas, 1986, Kelly *et al.,* 2018). The East China Sea is a region with a large marine traffic, and any pollution incidents can quickly and widely spread because of the presence of the Kuroshio Current. These can then reach environmentally important marine areas such as the coral reefs near the Ryuku Island Chain (Qiao *et al*., 2019; NOC, 2018). The objective here is to simulate virtual trajectories from data available from oil spill events during these two years. For the first case we base our simulations on the Golden Trader oil spill and for the second one, on the Sanchi oil spill. We try to find out how accurately we can represent the oil spill trajectories in these two regions. The trajectories are calculated at 15m depth (only depth at which the input velocity data is available).

1.3. Document structure

In addition to this introduction, this document includes the following chapters:

- Chapter 2 : Lagrangian drift of *Sargassum*
- Chapter 3 : Lagrangian drift of oil spills
- Chapter 4 : General conclusions and perspectives

1.4. Applicable & Reference documents

- Bobra AM and MF Fingas (1986) The Behaviour and Fate of Arctic Oil Spills. Water Sci Technol, 18, 13–23. doi: https://doi.org/10.2166/wst.1986.0012

- Chen, L., Yang, J., & Wu, L. (2019). Modeling the dispersion of dissolved natural gas condensates from the Sanchi incident. Journal of Geophysical Research: Oceans, 124(11), 8439-8454. https://doi.org/10.1029/2019JC015637

- Delandmeter, P., & Sebille, E. V. (2019). The Parcels v2. 0 Lagrangian framework: new field interpolation schemes. Geoscientific Model Development, 12(8), 3571-3584. https://doi.org/10.5194/gmd-12-3571-2019

- Kelly, S., Popova, E., Aksenov, Y., Marsh, R., & Yool, A. (2018). Lagrangian modeling of Arctic Ocean circulation pathways: Impact of advection on spread of pollutants. Journal of Geophysical Research: Oceans, 123, 2882- 2902. https://doi.org/10.1002/2017JC013460

- Miron, P., Olascoaga, M. J., Beron-Vera, F. J., Putman, N. F., Triñanes, J., Lumpkin, R., & Goni, G. J. (2020). Clustering of Marine Debris and Sargassum-Like Drifters Explained by Inertial Particle Dynamics. *Geophysical Research Letters*, *47*(19), e2020GL089874. https://doi.org/10.1029/2020GL089874

- National Oceanography Center (NOC), Southampton, UK, 2018b National Oceanography Center (NOC), Southampton, UK. Coral reefs may be at risk from Sanchi oil tanker contamination http://noc.ac.uk/news/coral-reefs-may-be-risk-sanchi-oil-tanker-contamination

- Oxenford, H.A.; Cox, S.-A.; van Tussenbroek, B.I.; Desrochers, A. Challenges of Turning the Sargassum Crisis into Gold: Current Constraints and Implications for the Caribbean. Phycology 2021, 1, 27–48. https://doi.org/10.3390/phycology1010003

- Pan, Q., Yu, H., Daling, P. S., Zhang, Y., Reed, M., Wang, Z., ... & Zou, Y. (2020). Fate and behavior of Sanchi oil spill transported by the Kuroshio during January–February 2018. Marine Pollution Bulletin, 152, 110917. https://doi.org/10.1016/j.marpolbul.2020.110917

- Pålsson, J., Hildebrand, L., & Lindén, O. (2017). Comparing Swedish oil spill preparedness to regional countries using the RETOS™ evaluation tool. In *International Oil Spill Conference Proceedings* (Vol. 2017, No. 1, pp. 21-36). International Oil Spill Conference. https://doi.org/10.7901/2169-3358-2017.1.21

- Qiao, F., Wang, G., Yin, L., Zeng, K., Zhang, Y., Zhang, M., ... & Chen, G. (2019). Modelling oil trajectories and potentially contaminated areas from the Sanchi oil spill. Science of the Total Environment, 685, 856-866. https://doi.org/10.1016/j.scitotenv.2019.06.255

- Rio, M.-H., S. Mulet, and N. Picot (2014). Beyond GOCE for the ocean circulation estimate: Synergetic use of altimetry, gravimetry, and in situ data provides new insight into geostrophic and Ekman currents, Geophys. Res. Lett, 41. https://doi.org/10.1002/2014GL061773

- Ubelmann, C. Algorithm Theoretical Basis Document for Total surface current at 0m and 15m: https://docs.google.com/document/d/1qVwZfxmd9iQF2YL1ZYGjmgVqo0oSnqUu_HCWmQ94fDk/edit

- Van Sebille, E., Zettler, E., Wienders, N., Amaral-Zettler, L., Elipot, S., & Lumpkin, R. (2021). Dispersion of surface drifters in the Tropical Atlantic. *Frontiers in Marine Science*, *7*, 607426. https://doi.org/10.3389/fmars.2020.607426

- Wang, M., Hu, C., Barnes, B. B., Mitchum, G., Lapointe, B., & Montoya, J. P. (2019). The great Atlantic Sargassum belt. Science, 365(6448), 83–87. https://doi.org/10.1126/science.aaw7912

1.5. Terminology

ATBD	Algorithm Theoretical Basis Document
CMEMS	Copernicus Marine Environment Monitoring Service
ESA	European Space Agency
EU	European Union
PUB	Publication
PUM	Product User Manual
RB	Requirement Baseline

2 Lagrangian drift of *Sargassum*

2.1. Overview

The Lagrangian drift of *Sargassum* is simulated using the OceanParcels framework (www.OceanParcels.org). OceanParcels is an open-source software, which has been developed by Utrecht University (Netherlands) to simulate the dispersion of plastic material in the ocean. The frontend of the OceanParcels framework is written in python, and has a back-end written in C for speed and efficiency. The Parcels code (**P**robably **A** **R**eally **C**omputationally **E**fficient **L**agrangian **S**imulator) consists of a set of Python classes and methods to create customisable particle tracking simulations using velocities from for example outputs from Ocean Circulation models. Version 2.3 of Parcels is used here (Delandmeter & van Sebille, 2019; https://github.com/OceanParcels/parcels/releases).

The processing followed in the generation of the data is shown in fig. 1.

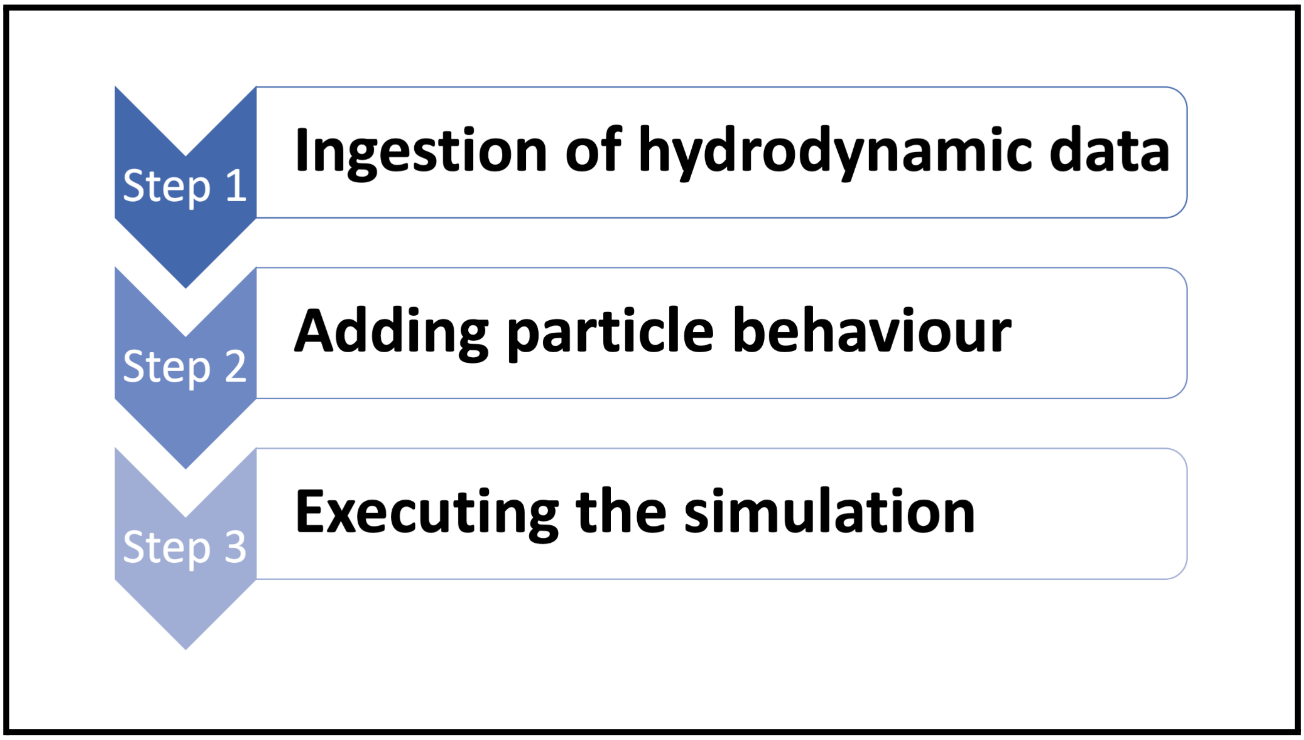


Figure 1. Flowchart describing the processing followed to generate the Surface Lagrangian drift simulations of Sargassum.

2.2. Step 1 : Input data

2.2.1. Ingestion of hydrodynamic data

The ocean parcels software uses the velocity fields to advect particles and simulate their trajectories. The velocity field inputs can be in different formats, and here we use netCDFs. The velocity fields used are a combination of satellite and drifter data. The data is available from 2010 to 2020, has a spatial resolution of 1/4°, an hourly temporal resolution and is available at 15m depth. The spatial resolution of 1/4° is due to the fact that the satellite geostrophic velocities from CMEMS and the ERA5 wind data have this spatial resolution, and it also provides a light product which can have a greater spatial coverage. Here we focus on velocity data from the Tropical Atlantic region in the years 2018 and 2019, for case 1 and 2, respectively. For more information on the input data see:
https://docs.google.com/document/d/1qVwZfxmd9iQF2YL1ZYGjmqVqo0oSnqUu_HCWmQ94fDk/edit .

2.2.2. Particle release location data

2.2.2.1. Case 1 : drifters scenario

Virtual particles are released in the Tropical Atlantic, following the release locations and times of the Miron *et al* (2020) field study. Fig. 2 shows the 4 release locations of the field study, and at each release location, different types of drifters are released, but we focus only on 3 of them (the other drifters are of different shape types, not as relevant for this study):

- drogued drifter (type 6) : Conventional Surface Velocity Program (SVP) drifters. These drifters have a spherical surface float with a holey sock, otherwise called drogue, attached to it and centered at a depth of 15m. At each location only one of these drifters is released.

- <u>undrogued drifter (type 0)</u> : Conventional SVP drifter, but without a drogue. At each location only one of these drifters is released.
- <u>*sargassum*-like drifter (type 1)</u> : Undrogued drifter, designed for the field study used in Miron *et al* (2020), to imitate floating, small patches of pelagic *Sargassum*. They are made to look like *Sargassum*-like mats by using an artificial boxwood hedge to mimic their appearance (see Miron *et al.* 2020 and references therein for illustrations and further details.) At each location three of these drifters are released. (Miron *et al*., 2020)

Figure 2 shows drifter data from Miron *et al.* (2020). There are 4 release locations : d1, d2, d3 and d4. Only the trajectories from the 3 drifter types mentioned above are shown. It is important to notice that at each release location, only 1 undrogued (black) and 1 drogued (red) drifter were released, but 3 *Sargassum*-like (blue) drifters were released.

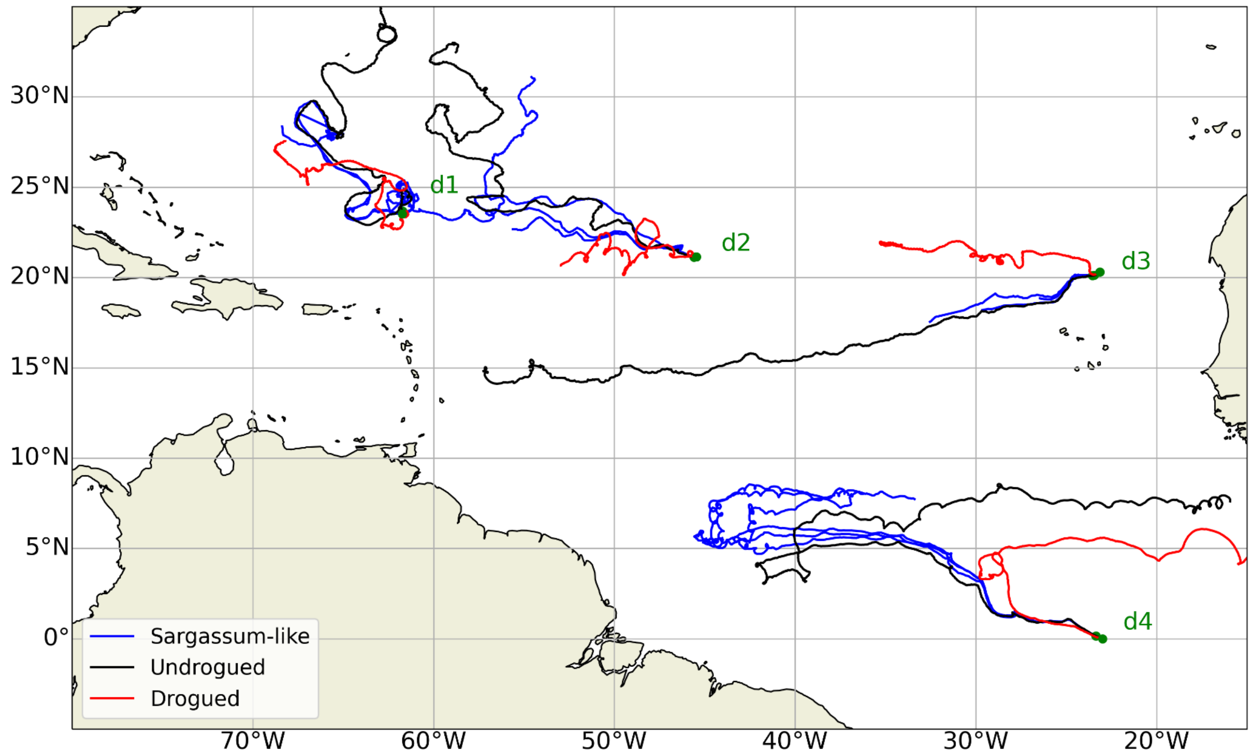


Figure 2. The trajectories of type 0 (undrogued), 1 (Sargassum-like) and 6 (drogued) drifters from Miron et al (2020). Green dots indicate the release locations. One undrogued, one drogued and three Sargassum-like drifters are released at every location (d1, d2, d3 and d4). The black box indicates the WOC Tropical Atlantic domain.

2.2.2.2. Case 2 : satellite scenario

Virtual particles are also released in the Tropical Atlantic during May 2019. The release locations are based on the identification of *Sargassum* meshes in satellite images by MeteoFrance. The location of the meshes is then identified after 24 hours. These start and end locations are shown in fig. 3.

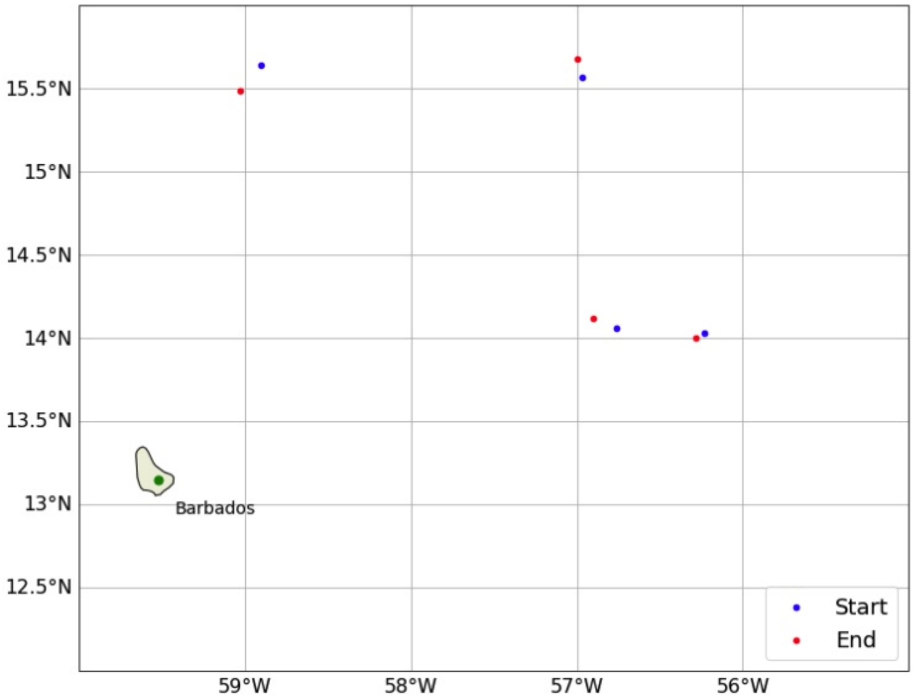


Figure 3. Start (blue) and end (red) locations of Sargassum mats observed via satellite imagery during May 2019 by MeteoFrance.

2.3. Step 2 : Adding particle behaviour

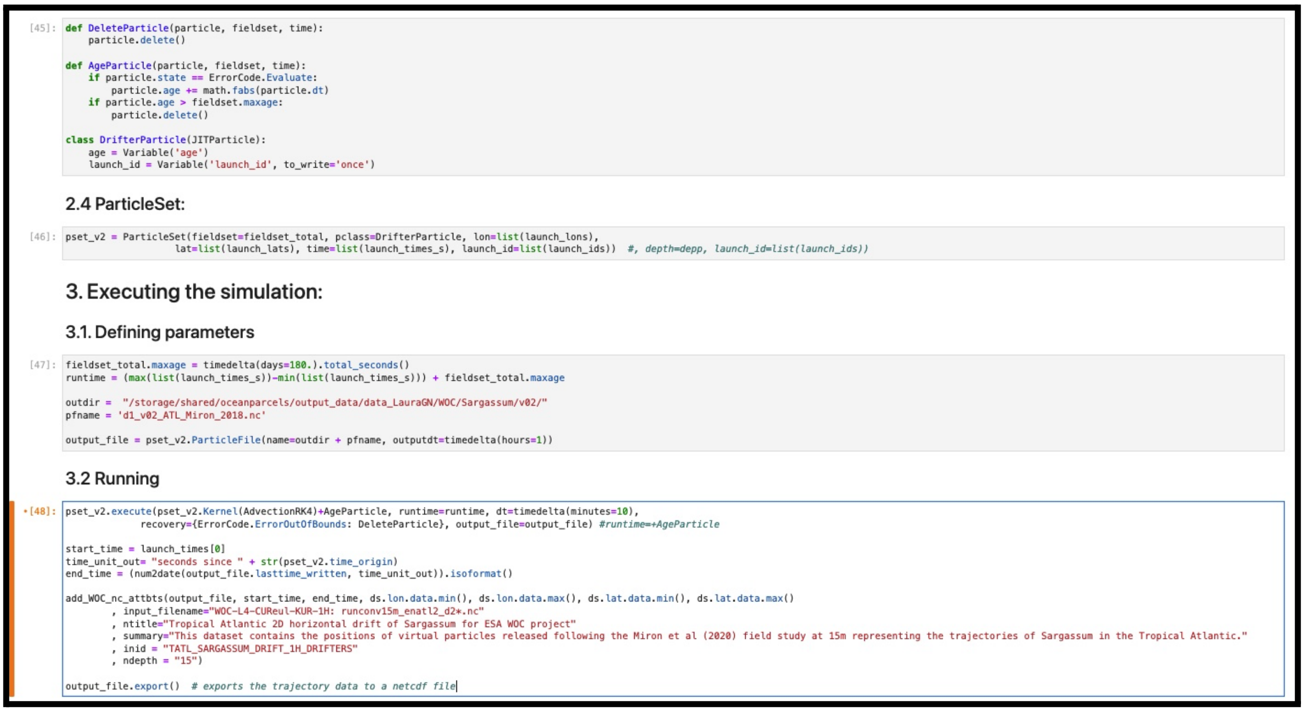
2.3.1. Description

The particles' (*Sargassum* in this case) behaviour is represented here as a Lagragian drift by horizontally advecting them (in 2D) using the ingested velocity fields. The velocity fields used here include geostrophic currents, inertial oscillations and Ekman transport. In this second version (v2.0) preliminary simulation of the product, we assume the *Sargassum* follows the total velocity of the new currents product at 15m. This allows us to first understand the particles' trajectories with this new velocity field, and in next versions the surface dynamics (wind effects and Stokes drifts) will be added as well as the *Sargassum* particles' behaviour.

2.3.2. Kernel set-up

The main kernel is the horizontal advection of particles. The new velocity product includes two components: a geostrophic and an ageostrophic (Ekman transport and inertial oscillations) one. These two velocity fieldsets are included in the advection kernel. Particles which reach the ocean boundaries of the domain are deleted using the DeleteParticle function. For the *Sargassum* drifters simulation, an additional kernel is created : the AgeParticle kernel. As the particles are released at different times, it allows to keep track of the age of the particles released on different dates and times, and to simulate their trajectories for the same amount of days.

Trajectories are integrated using a two-dimensional fourth-order Runge-Kutta scheme with an integration time step of 10 minutes. The 2D position of each particle is stored every hour. A jupyter-notebook example of the code is shown in fig. 4.



```
[45]: def DeleteParticle(particle, fieldset, time):
          particle.delete()

      def AgeParticle(particle, fieldset, time):
          if particle.state == ErrorCode.Evaluate:
              particle.age += math.fabs(particle.dt)
          if particle.age > fieldset.maxage:
              particle.delete()

      class DrifterParticle(JITParticle):
          age = Variable('age')
          launch_id = Variable('launch_id', to_write='once')
```

2.4 ParticleSet:

```
[46]: pset_v2 = ParticleSet(fieldset=fieldset_total, pclass=DrifterParticle, lon=list(launch_lons),
                         lat=list(launch_lats), time=list(launch_times_s), launch_id=list(launch_ids))  #, depth=depp, launch_id=list(launch_ids))
```

3. Executing the simulation:

3.1. Defining parameters

```
[47]: fieldset_total.maxage = timedelta(days=180.).total_seconds()
      runtime = (max(list(launch_times_s))-min(list(launch_times_s))) + fieldset_total.maxage

      outdir =  "/storage/shared/oceanparcels/output_data/data_LauraGN/WOC/Sargassum/v02/"
      pfname = 'd1_v02_ATL_Miron_2018.nc'

      output_file = pset_v2.ParticleFile(name=outdir + pfname, outputdt=timedelta(hours=1))
```

3.2 Running

```
•[48]: pset_v2.execute(pset_v2.Kernel(AdvectionRK4)+AgeParticle, runtime=runtime, dt=timedelta(minutes=10),
                    recovery={ErrorCode.ErrorOutOfBounds: DeleteParticle}, output_file=output_file) #runtime=+AgeParticle

       start_time = launch_times[0]
       time_unit_out= "seconds since " + str(pset_v2.time_origin)
       end_time = (num2date(output_file.lasttime_written, time_unit_out)).isoformat()

       add_WOC_nc_attbts(output_file, start_time, end_time, ds.lon.data.min(), ds.lon.data.max(), ds.lat.data.min(), ds.lat.data.max()
               , input_filename="WOC-L4-CUReul-KUR-1H: runconv15m_enatl2_d2*.nc"
               , ntitle="Tropical Atlantic 2D horizontal drift of Sargassum for ESA WOC project"
               , summary="This dataset contains the positions of virtual particles released following the Miron et al (2020) field study at 15m representing the trajectories of Sargassum in the Tropical Atlantic."
               , inid = "TATL_SARGASSUM_DRIFT_1H_DRIFTERS"
               , ndepth = "15")

       output_file.export()  # exports the trajectory data to a netcdf file
```

Figure 4. Example of kernel set-up for the execution of the drifters Sargassum simulation at location d1.

2.4. Step 3 : Executing the simulation

2.4.1. Case 1 : drifters scenario

Once the hydrodynamic input data and particle behaviour kernels are ready, the simulations can be run and saved as netCDFs. To do so, a set of parameters need to be decided:

- **Timestep (dt)** : 10 minutes
- **Output timestep (dt) :** Hourly
- **Simulation length:** 180 days
- **Initial positions and release dates :** Release locations and times following Miron *et al*. (2018). Fig. 5 shows the release locations of the particles at the 4 sites of the Miron *et al*. (2020) field study. At each point 3 drifters are released of the type sargassum-like, one undrogued and one drogued drifter.

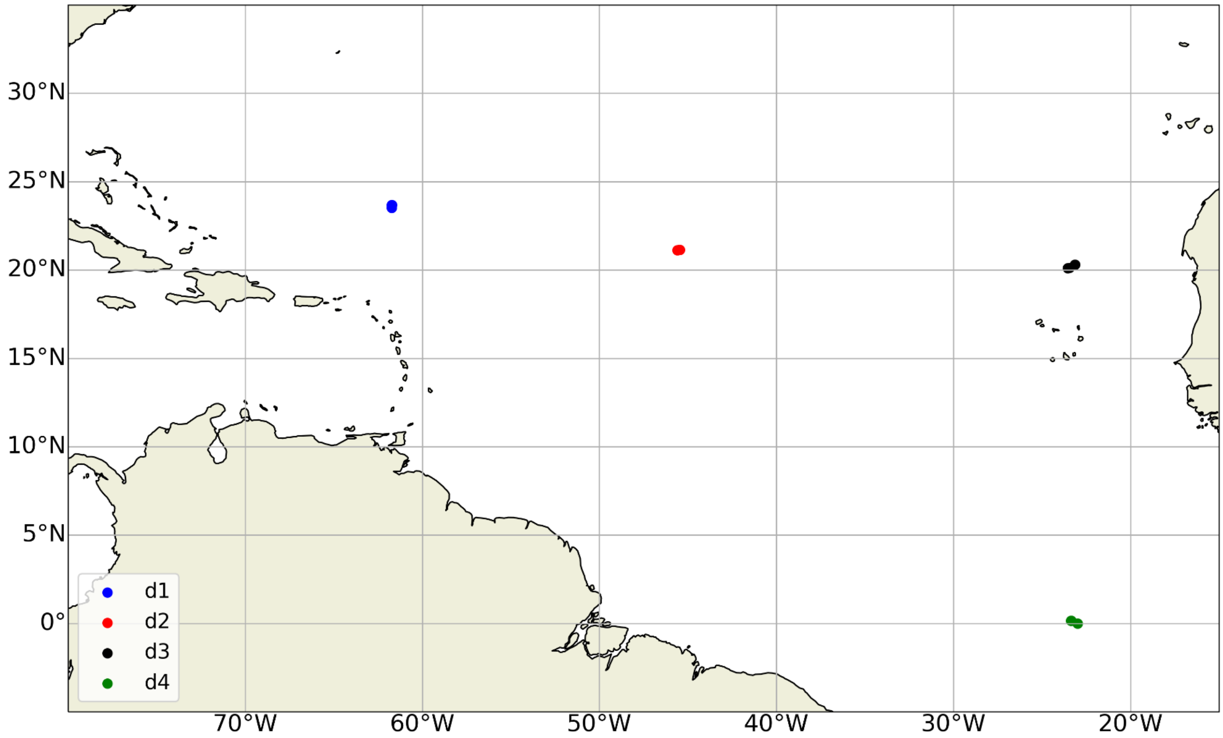


Figure 5. Release locations of the particles following Miron et al. (2020).

2.4.2. Case 2 : satellite scenario

Once the hydrodynamic input data and particle behaviour kernels are ready, the simulations can be run and saved as netCDFs. To do so, a set of parameters need to be decided:

- **Timestep (dt)** : 10 minutes
- **Output timestep (dt) :** Hourly
- **Simulation length:** 1 day
- **Initial positions and release dates :** Release locations are set where *Sargassum* meshes have been identified in satellite images (information provided by MeteoFrance). Particles are released on a regular grid (mesh) with a spacing of 0.02° around the satellite image location (fig. 6).

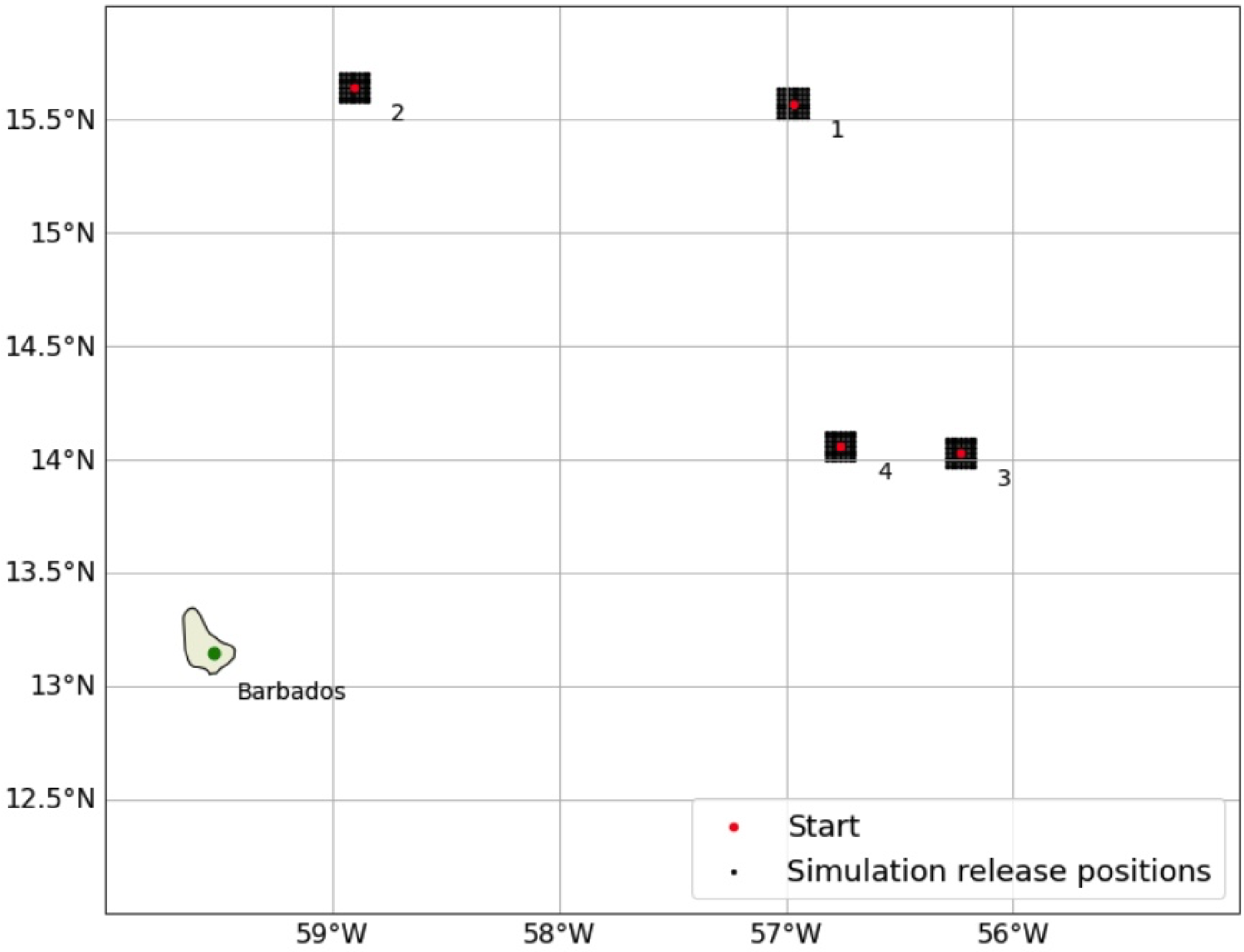


Figure 6. Release locations of the virtual Sargassum particles (black) around the 4 locations where Sargassum mats have been identified on the satellite image (red).

2.5. Product validation

The second version of these *Sargassum* drift simulations has been generated with the second version (v02) of the velocity product. Therefore, like for version 1 of this product, this first simulation with v02 velocities will allow us to identify how and where improvements are needed. The drifter surface Lagrangian drift of Sargassum will be validated using the drifter trajectories of Miron *et al.* (2020). The simulated trajectories of *Sargassum* are also compared to the CMEMS fields on which the new velocity products are based and to version 01 of this product.

2.5.1. Data

Our simulations are also compared with ones done using as input a CMEMS-available ocean velocity dataset. We choose the same dataset to extract the geostrophic velocity to create the new velocity product used here (the WOC ocean velocity v02, for further details see https://docs.google.com/document/d/1qVwZfxmd9iQF2YL1ZYGjmgVqo0oSnqUu_HCWmQ94fDk/edit). The dataset is the Global Total Surface and 15m Current (COPERNICUS-GLOBCURRENT) from Altimetric Geostrophic Current and Modeled Ekman Current Reprocessing (MULTIOBS_GLO_PHY_REP_015_004; Rio *et al*, 2014). It is obtained from the CMEMS site : https://resources.marine.copernicus.eu/product-detail/MULTIOBS_GLO_PHY_REP_015_004/INFORMATION. This product provides zonal and meridional horizontal ocean velocity components at 0m and 15m depths. It has a global coverage and a spatial resolution of 1/4°. We use the outputs with a temporal resolution of 3 hours.

2.5.1.1. Case 1 (drifters scenario)

The drifter surface Lagrangian drift of *Sargassum* is validated using the drifter trajectories of Miron *et al.* (2020). As explained in Miron *et al* (2020) different types of drifters are released. Here we focus on 3 types: type 0 (undrogued (Stokes) drifter), type 1 (*Sargassum*-like drifter) and type 6 (SVP (drogued) drifter). See section 2.2.2.1 for further details.

2.5.1.2. Case 2 (satellite scenario)

The simulations are compared to the information provided by meteofrance on the satellite-identified *Sargassum* meshes. The start and end locations (fig. 3), and the dates of end and start locations are provided.

In this case we cannot compare to the v01 velocity fields, as the data is not available for 2019.

2.5.2. Results

2.5.2.1. Case 1 (drifters scenario)

Fig. 7 shows all the simulated trajectories of the *Sargassum* drifters after 180 days at 15m. At location d4 the trajectories are very short as the virtual particles flow south and leave the domain. Most trajectories move westward as expected and also obtained in previous studies such as van Sebille *et al*. (2021).

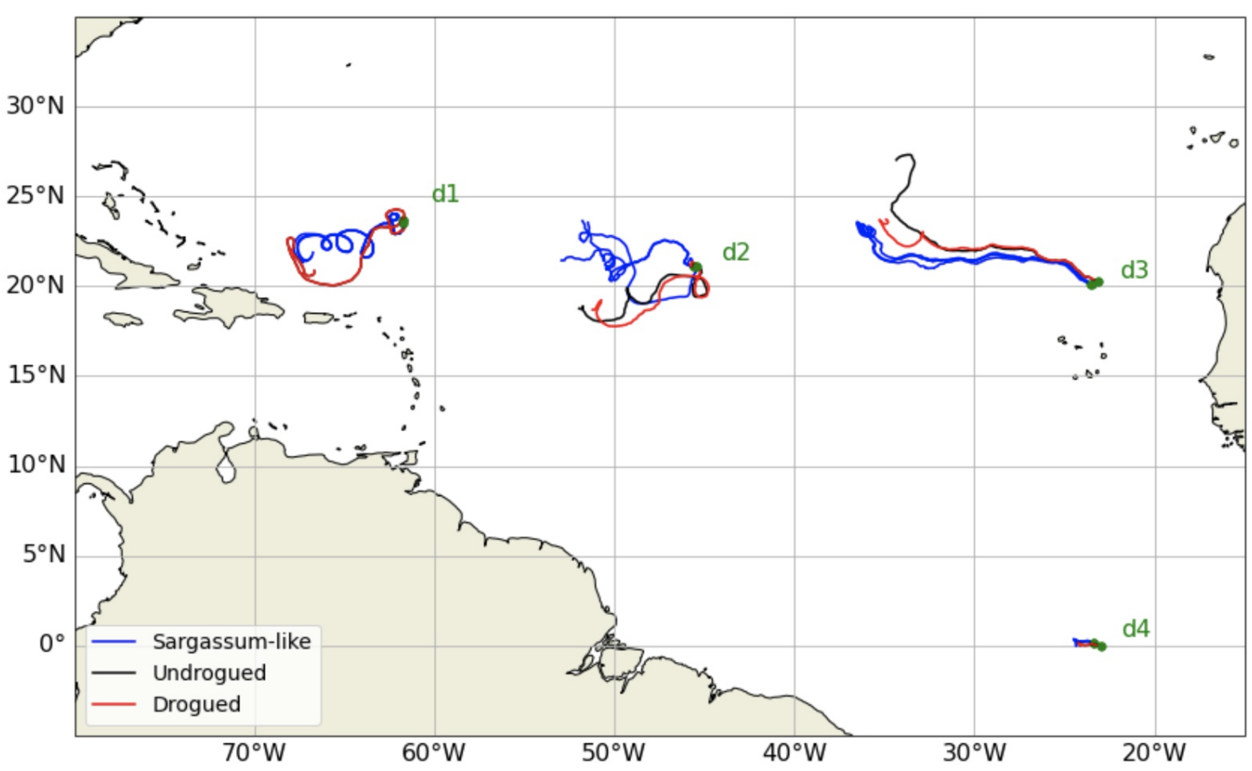


Figure 7. Simulated Sargassum-like drifters' trajectories after 180 days at 15m. Three different drifters are released at each location and each of them is represented in a different colour (Sargassum-like in blue, Undrogued in black and drogued in red). Green dots show the starting points.

Figures 8 to 11 show the trajectories of the real drifters and of the virtual drifters simulated from the WOC data. At location d1, an eddy seems to be present that traps the drifters. This causes big differences between the real and the simulated trajectory. The real drifters also go more northward than the simulated ones. At location d2, though the trajectories are not very close, both of them go westward. Also, although the WOC product starts to resolve inertial oscillations, more of these can be observed in the real drifters than the simulated drifters. At location d3, again, the simulated drifters move westward like the real drifters. In this case the simulated drifters go more northward than the real ones. The simulated drifters perform best when compared to the drogued drifter. This is expected as for this version of the product, the virtual drifters are only advected with the fields at 15m. Lastly, same at location d4, the simulated drifters move westward too, but they go more southward than the real drifters and leave the southern domain of the velocity data (the equator).

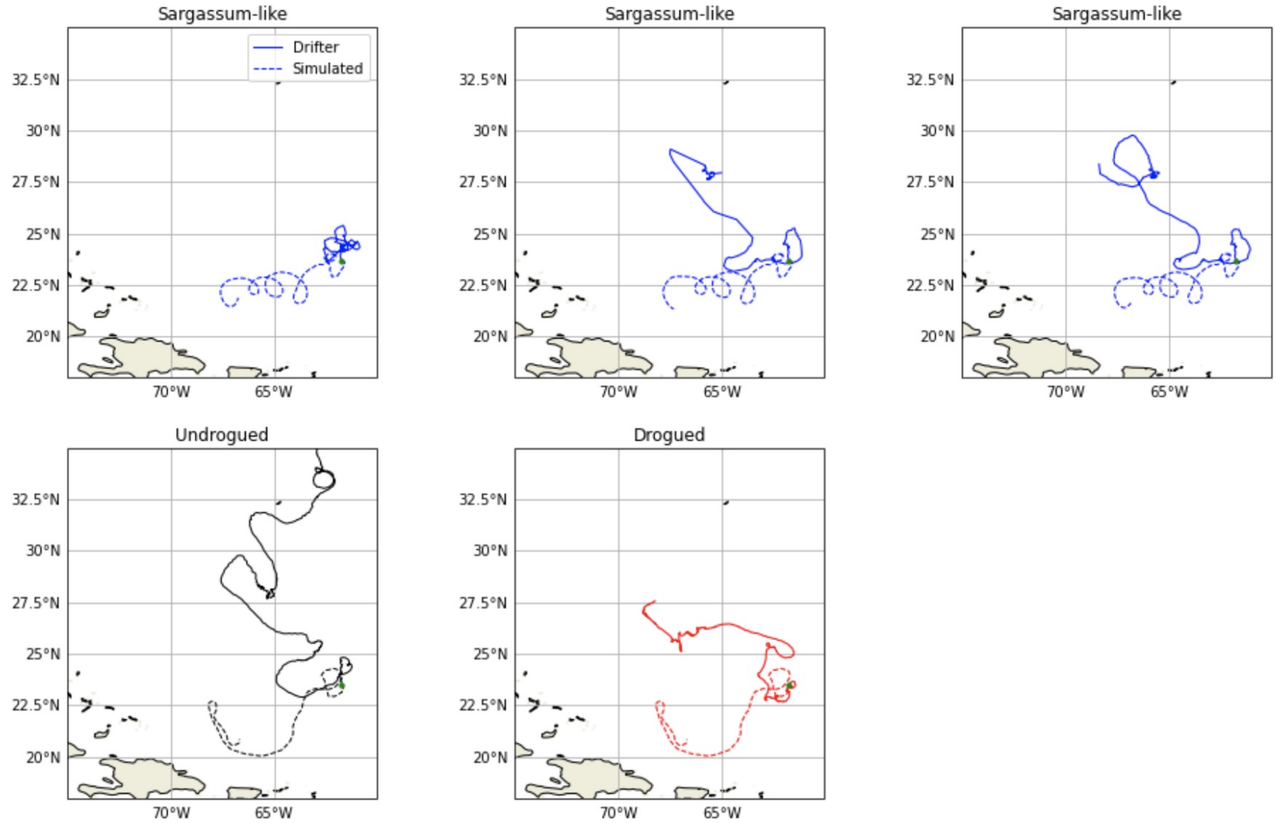


Figure 8. Trajectories of the Miron et al. (2020) drifters (bold) and the simulated WOC trajectories (dashed) at location d1. Green dot shows the release location.

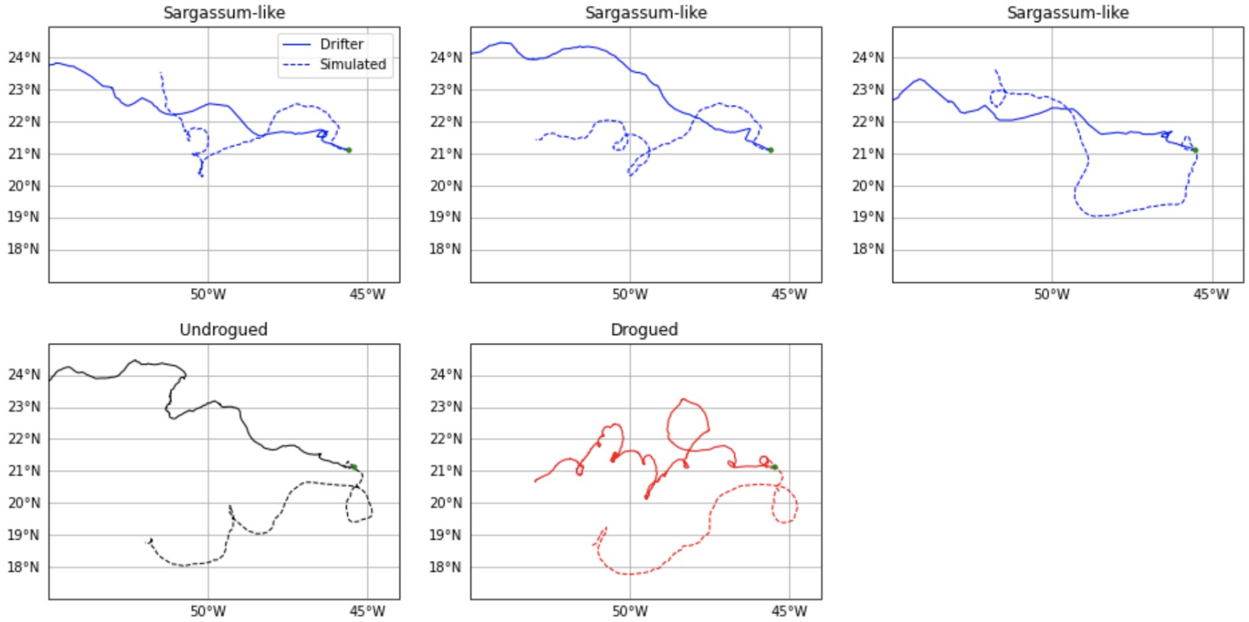


Figure 9. Trajectories of the Miron et al. (2020) drifters (bold) and the simulated WOC trajectories (dashed) at location d2. Green dot shows the release location.

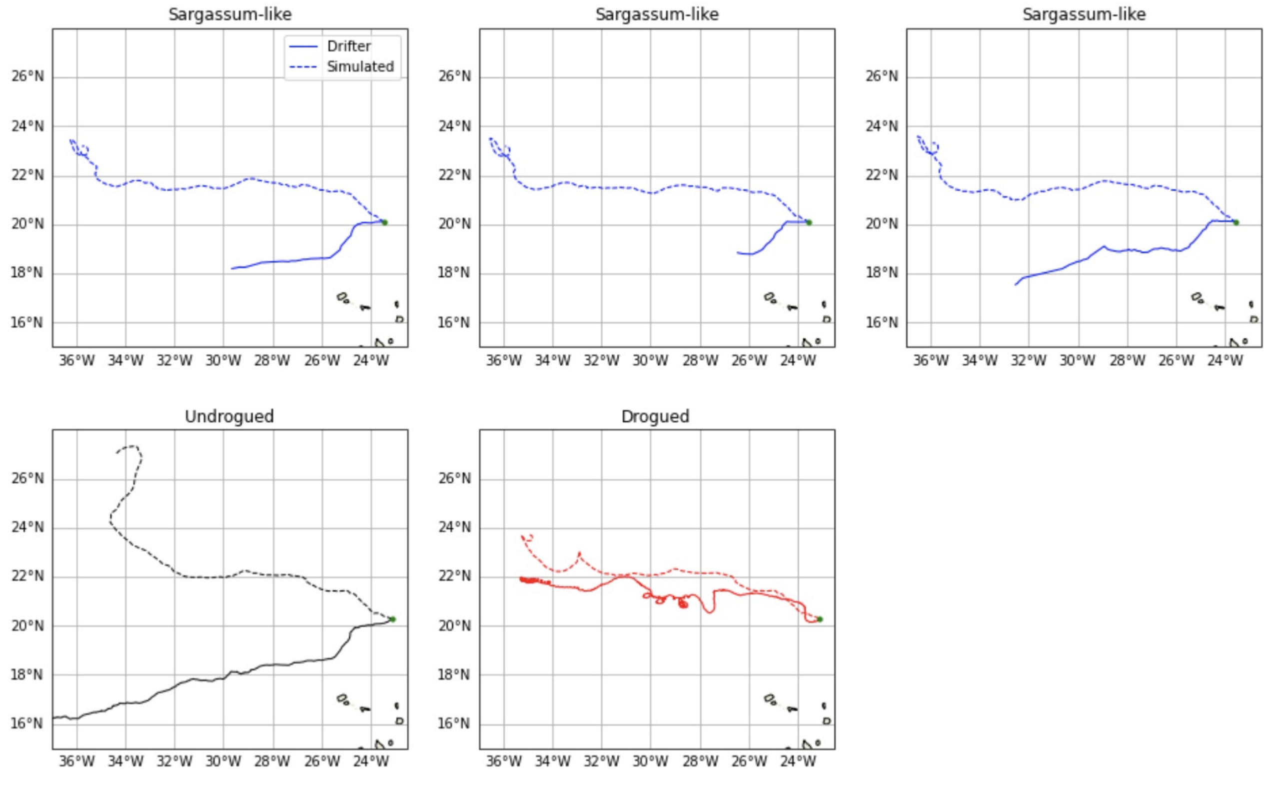


Figure 10. Trajectories of the Miron et al. (2020) drifters (bold) and the simulated WOC trajectories (dashed) at location d3. Green dot shows the release location.

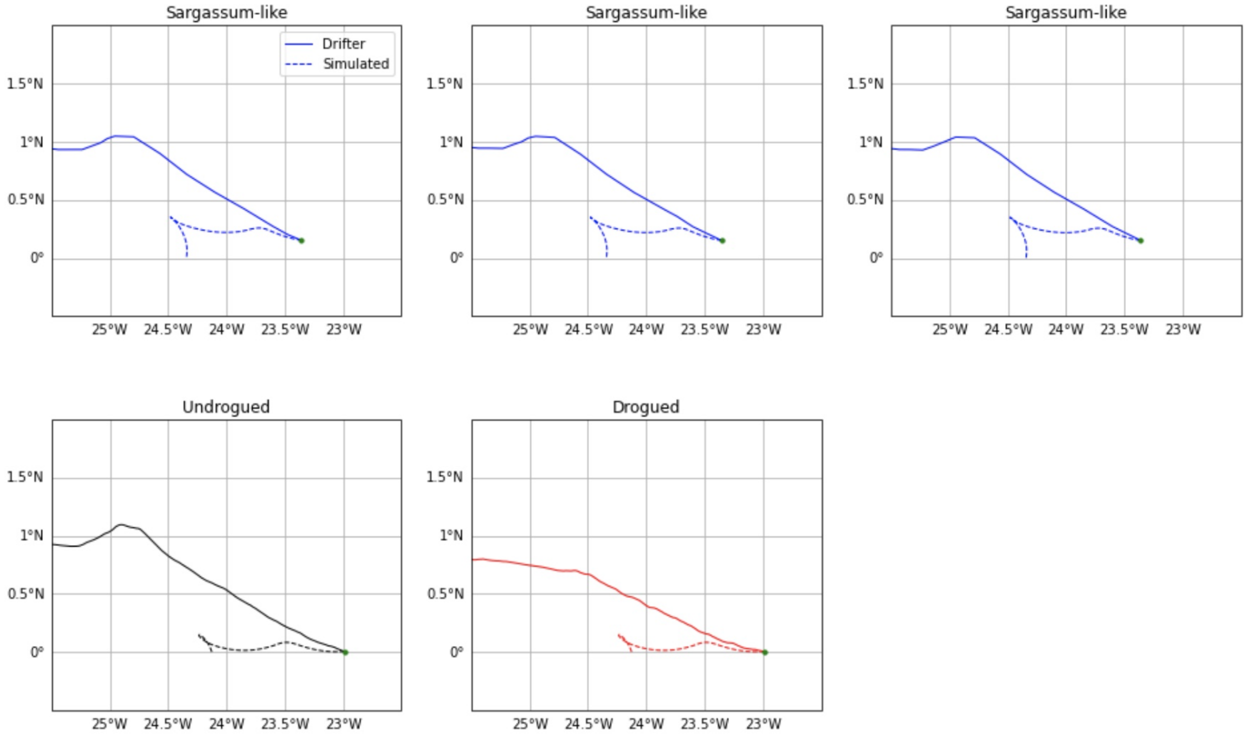


Figure 11. Trajectories of the Miron et al. (2020) drifters (bold) and the simulated WOC trajectories (dashed) at location d4. Green dot shows the release location.

2.5.2.2. Case 2 (satellite scenario)

We compare the simulated *Sargassum* mesh trajectories with the satellite-identified locations at th 4 locations. Figure 12 shows that only for mesh 2, we obtain an accurate simulation with only the WOC velocity fields at 15m are used.

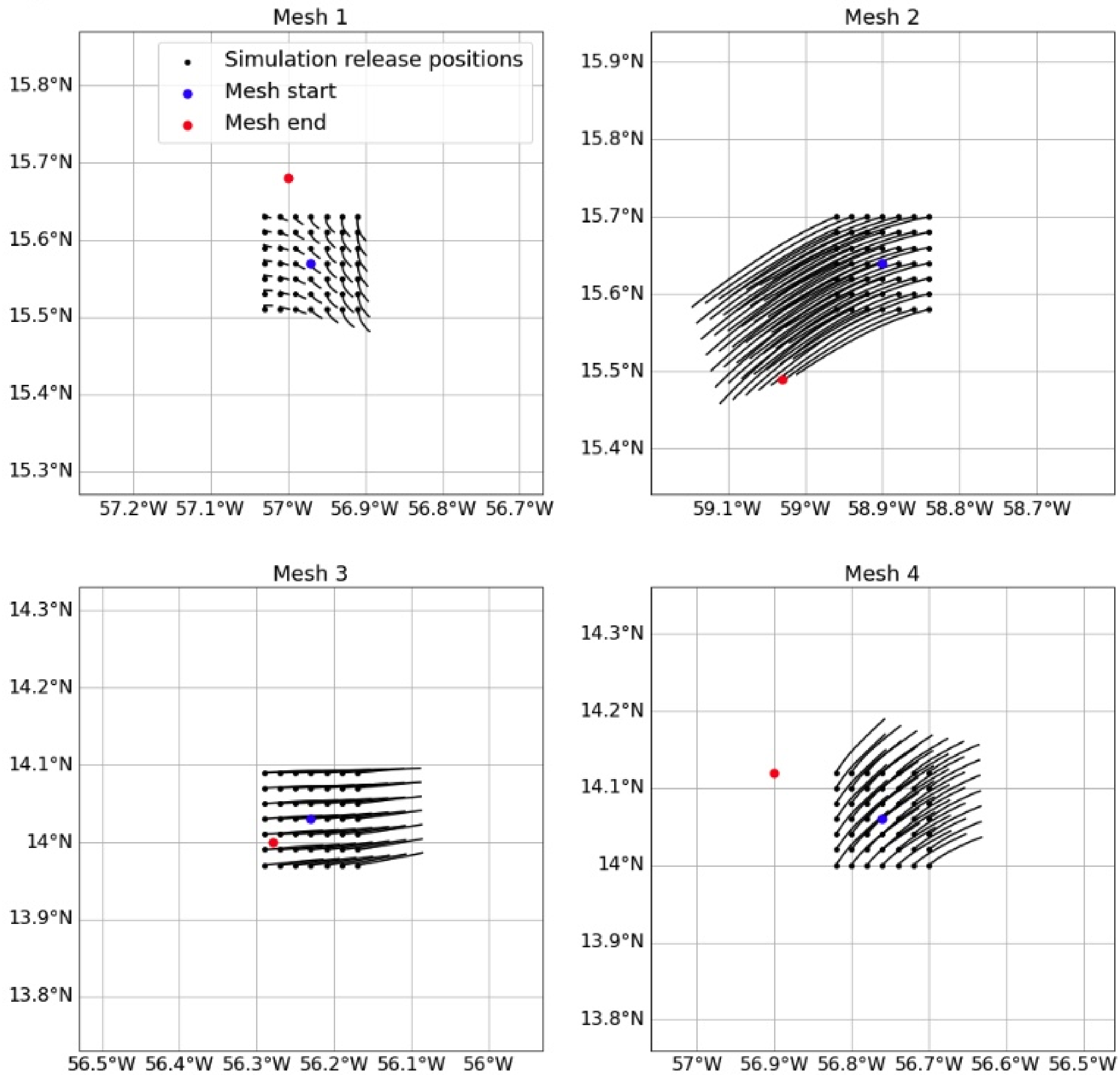


Figure 12. Simulation of the trajectories after 24 hours of each of the 4 satellite-identified Sargassum meshes (black lines). The black dots show the release locations of the simulation, the blue dots the start and the red dots the end locations of the satellite-identified Sargassum meshes.

In fig. 13 the total distance travelled by the *Sargassum* meshes is compared to the simulated trajectories. As observed above, the most accurate distance travelled is simulated for mesh 2, although it slightly exceeds the real data value. For meshes 1 and 4, the simulated meshes travel a much lower distance than the real meshes, especially for mesh 1. For mesh 4, WOC performs better than the simulations done with CMEMS at 15m velocities. Lastly, for mesh 3, the CMEMS simulation is more accurate than the WOC dataset, and is only slightly less than the real distance travelled.

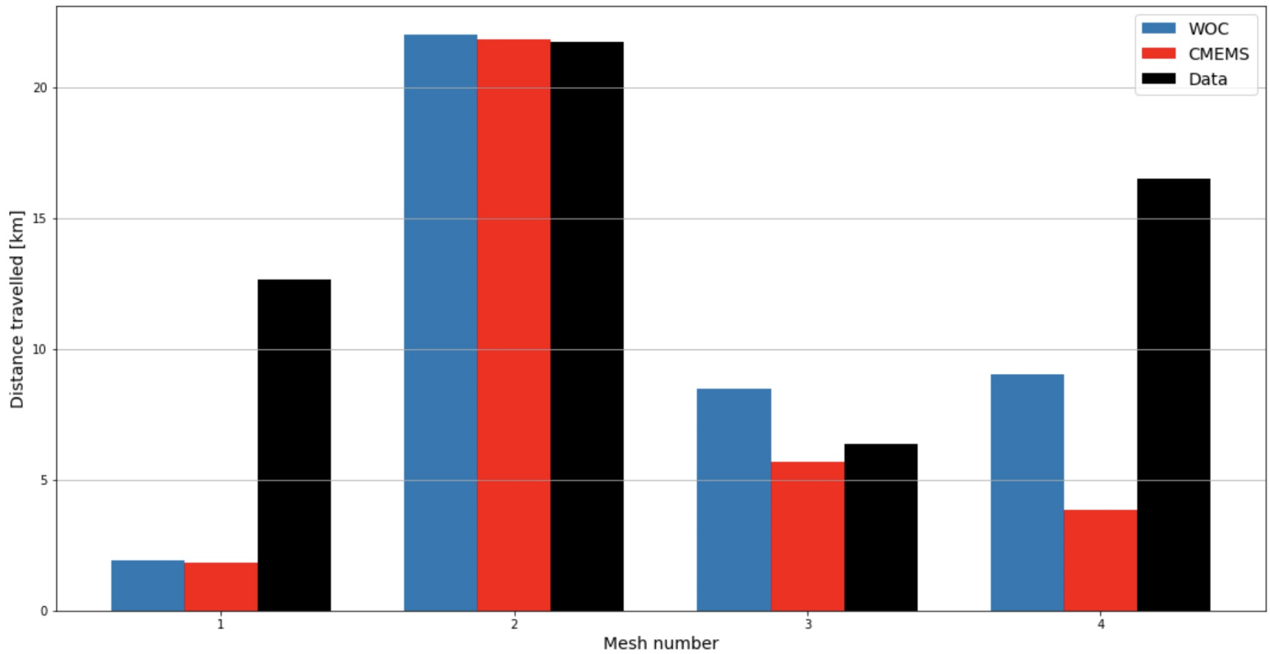


Figure 13. Comparison of the distance travelled by the satellite-identified Sargassum meshes (black) with the mean distance travelled by the simulated trajectories from the WOC (blue) and CMEMS at 15m (red) datasets.

Fig. 14 shows the separation between the satellite-identified meshes and the mean end location of the virtual particles. The separation distances for both the WOC and CMEMS simulations range between approximately 7 and 25 km. Except for mesh number 2, the CMEMS simulations are closer to the real end location of the satellite-identified meshes. Mesh number 2 is the one for which both datasets perform best, mesh number 4 the worst and the others very similar. The right plot of fig. 14 shows the same separation distances, but normalized by the total distance travelled by the satellite-identified meshes. The differences are relatively significant. In the best case, mesh number 2, the separation distance is half of the total distance travelled by the *Sargassum* meshes. With the separations normalized, the simulations for mesh number 3 yield the worst results, with the distances being around twice the distance travelled by the *Sargassum* meshes.

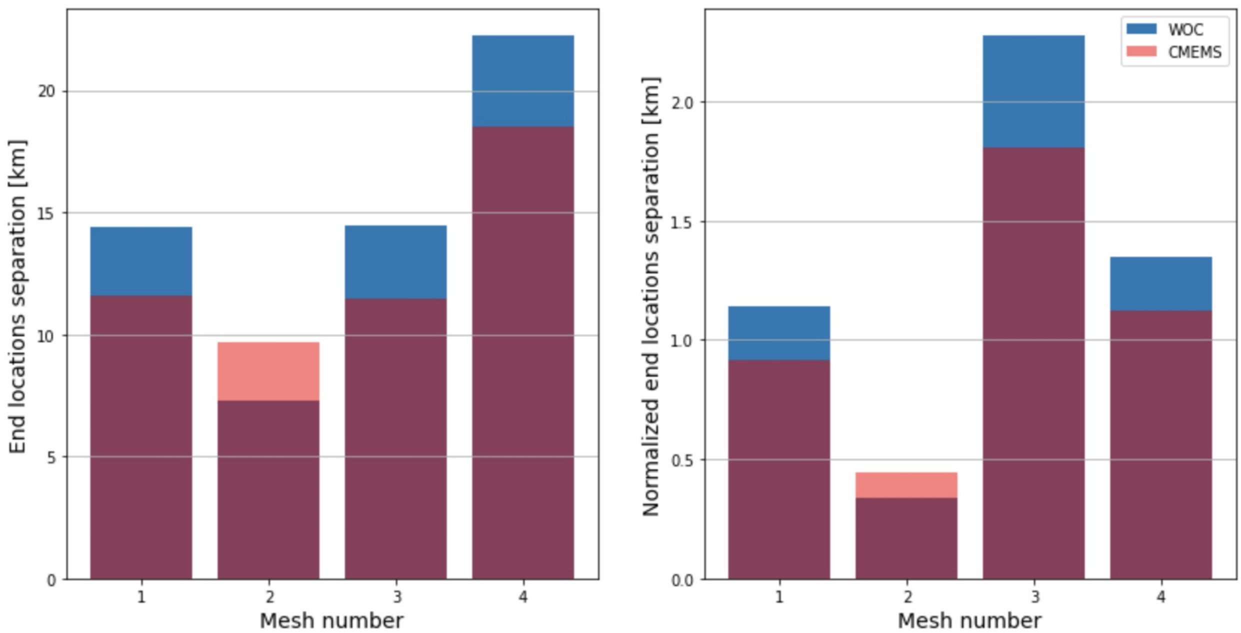


Figure 14. Separation in km between the satellite-identified meshes and the mean end location of the virtual particles (left). Separation normalized by the distance travelled by the satellite-identified meshes (right).

Lastly, fig. 15 shows the difference with the angle of the direction (in degrees) of the trajectory of the satellite-identified *Sargassum* meshes start and end locations. Again the best result is obtained for mesh 2, with the WOC dataset performing slightly better than CMEMS. For the other meshes, CMEMS performs better than WOC, especially for mesh 1. In this case WOC is nearly completely in the opposite direction, whilst the difference with the CMEMS simulations is only around 50 degrees. For meshes 3 and 4, both datasets perform badly with an angle difference of more than 100 degrees.

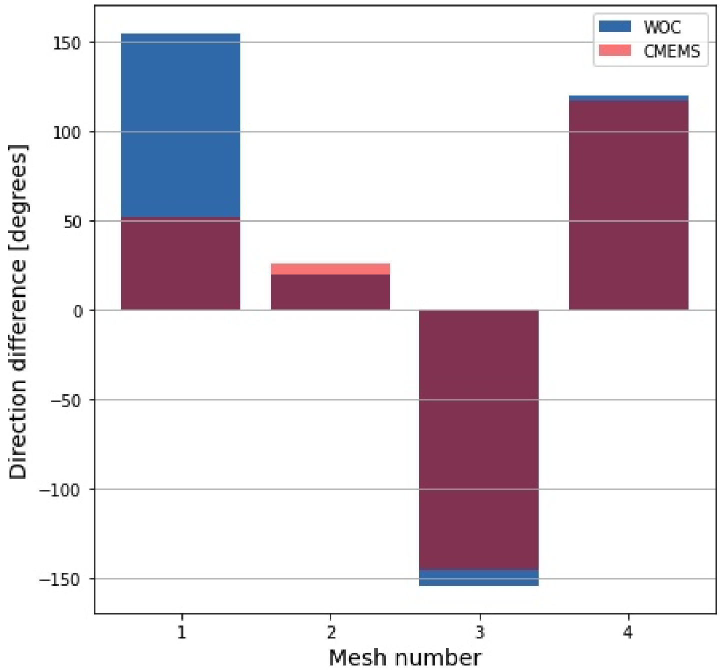


Figure 15. Difference with the angle of the direction (in degrees) of the trajectory of the satellite-identified *Sargassum* meshes start and end locations*.*

2.5.3. Conclusions and future work

Some promising results are obtained already with just the WOC fields at 15m. Adding the wind effects and/or the Stokes drift will likely improve the results. A further comparison with the CMEMS fields at 0m with both test cases will also be done. Further diagnostics are also being developed for the drifter scenario in order to quantitatively validate the simulations. The weather conditions during the different test cases also need to be analysed in order to understand the impact of for example the strength of the wind present on the performance of the simulations.

3. Surface Lagrangian drift of oil spills

3.1. Overview

The Lagrangian drift of oil is simulated using the OceanParcels framework (www.OceanParcels.org). OceanParcels is an open-source software, which has been developed by Utrecht University (Netherlands) to simulate the dispersion of plastic material in the ocean. The frontend of the OceanParcels framework is written in python, and has a back-end written in C for speed and efficiency. The Parcels code (**P**robably **A** **R**eally **C**omputationally **E**fficient **L**agrangian **S**imulator) consists of a set of Python classes and methods to create customisable particle tracking simulations using output from Ocean Circulation models. Version 2.3 of Parcels is used here (Delandmeter & van Sebille, 2019).

The processing followed in the generation of the data is shown in fig. 16.

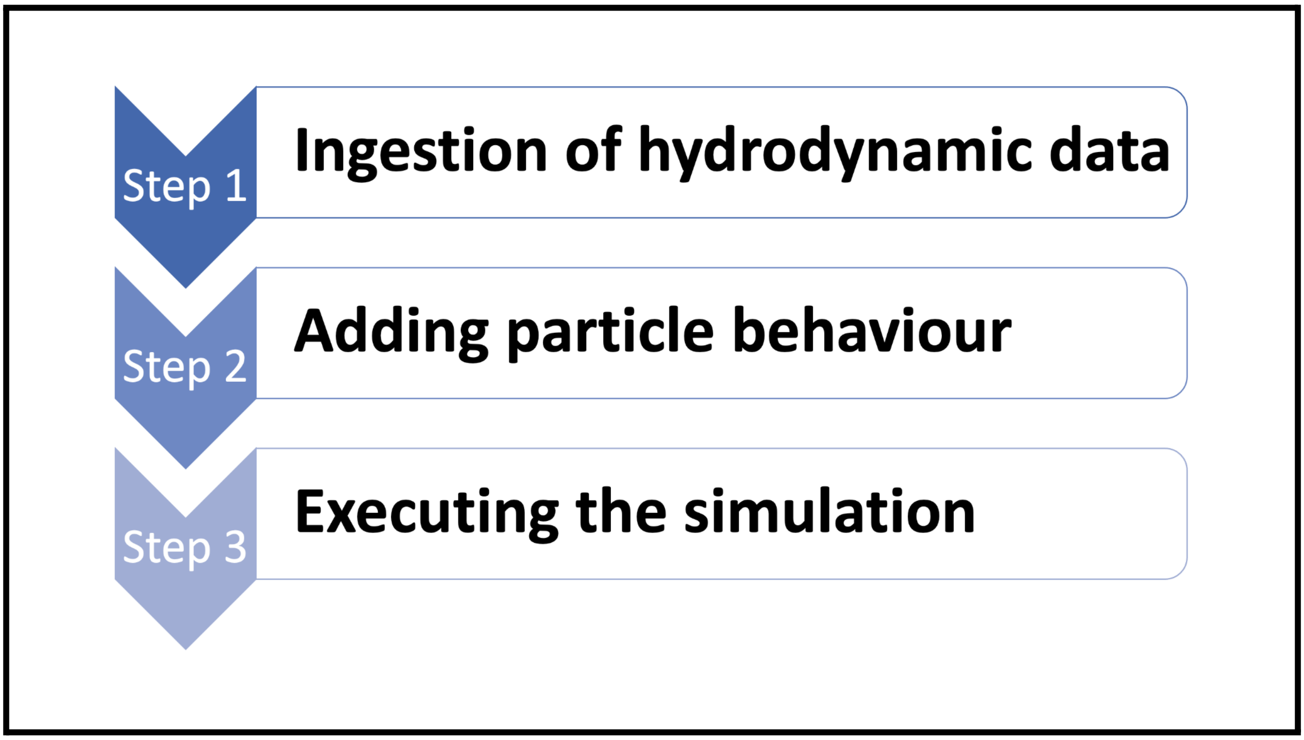


Figure 16. Flowchart describing the processing followed to generate the Surface Lagrangian drift simulations of oil.

3.2. Step 1 : Input data

3.2.1. Ingestion of hydrodynamic data

The ocean parcels software uses the velocity fields to advect particles and simulate their trajectories. The velocity field inputs can be in different formats, and here we use netCDFs. The velocity fields used are a combination of satellite and drifter data. The data is available from 2010 to 2020, has a spatial resolution of 1/4°, an hourly temporal resolution and is available at 15m depth. The spatial resolution of 1/4° is due to the fact that the satellite geostrophic velocities from CMEMS and the ERA5 wind data have this spatial resolution, and it also provides a light product which can have a greater spatial coverage. Here we focus on the velocity data generated for two regions : the Atlantic region and the Kuroshio Current region, for the years 2011 and 2018, respectively. For more information on the input data see : https://docs.google.com/document/d/1qVwZfxmd9iQF2YL1ZYGjmgVqo0oSnqUu_HCWmQ94fDk/edit .

3.2.2. Particle release locations

3.2.2.1. Case 1 : Golden Trader oil spill

Virtual particles were released in the North Atlantic during 2011. The release locations are based on the Golden Trader oil spill event. This oil spill incident occurred on the 10th of September 2011. A collision with a fishing vessel took place off the west coast of Denmark, ~40 km SW of Ringkobing Fjord. The substance spilt was bunker fuel (IFO), and the Swedish coast was impacted, more precisely the Swedish island of Tjörn (Pålsson *et al*., 2017). The oil spill is simulated for 14 days. The locations of the oil spill release and the regions of the Swedish coasts affected are shown in fig. 17. The first arrival of the oil was at the southern dots show on the 16th of September 2011, and it arrived on the northern part on the 21st of September 2011.

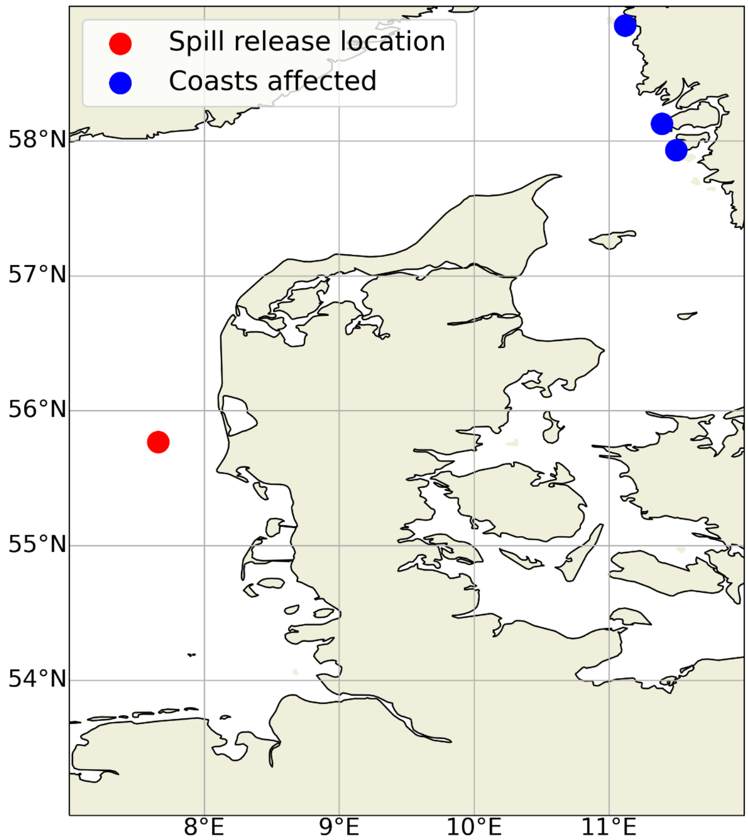


Figure 17. Locations of the oil spill event (red dot) and the regions of the Swedish coast affected (blue dots).

3.2.2.2. Case 2 : Sanchi oil spill

Virtual particles were released in the Kuroshio region during 2018. The release locations are based on the Sanchi oil spill event. The 6th of January 2018, Sanchi, an oil tanker, collided with a cargo ship in the East China Sea (Qiao *et al*., 2019). When Sanchi sank on 14th January 2018 (16:45 local time) HFO was spilled. The first island to be impacted was Takarajima (part of the Tokara Islands), at the end of January, on the 28th (Chen *et al*., 2019; Pan *et al*., 2020). The other island to be badly affected was Amami (and the adjacent southern islands) where oil was reported to land on the 1st of February (Pan *et al*., 2020). Contamination also affected Kikaijima and Tokunojima. The oil spill is simulated for a period of 16 days. The locations of the oil spill release and the coasts affected are shown in fig. 18.

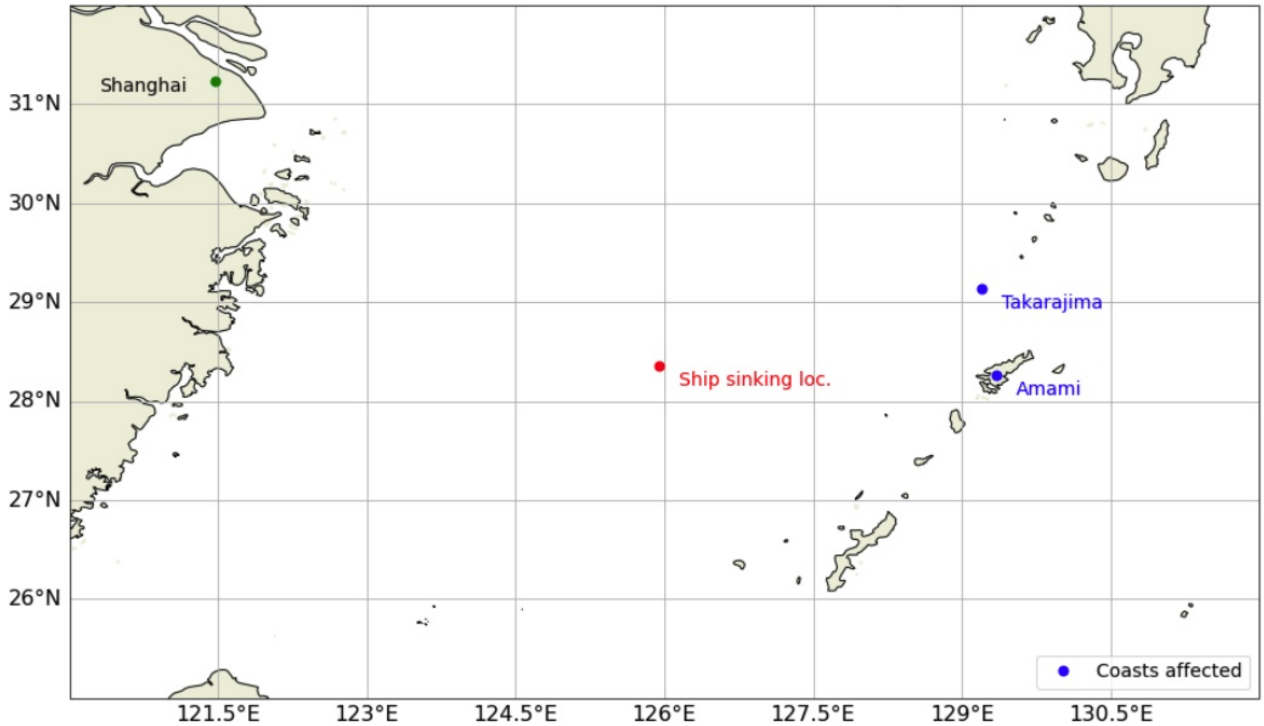


Figure 18. Locations of the oil spill event (red dot) and the regions first affected by the oil spill (blue dots).

3.3. Step 2 : Adding particle behaviour

3.3.1. Description

The particles' (oil in this case) behaviour is represented here as a Lagragian drift, by horizontally advecting them (in 2D) using the ingested velocity fields. The velocity fields used here include geostrophic currents, inertial oscillations and Ekman transport. In this second version (v2.0) preliminary simulation of the product, we assume the oil follows the total velocity of the new currents product at 15m. This allows us to first understand the particles' trajectories with this new velocity field, and in next versions the surface dynamics (wind effects and Stokes drifts) will be added as well as the oil particles' behaviour.

3.3.2. Kernel set-up

The main kernel is the horizontal advection of particles. The new velocity product includes two components: a geostrophic and an ageostrophic (Ekman transport and inertial oscillations) one. These two velocity fieldsets are included in the advection kernel. Particles which reach the ocean boundaries of the domain are deleted using the DeleteParticle function.

Trajectories are integrated using a two-dimensional fourth-order Runge-Kutta scheme with an integration time step of 10 minutes. The 2D position of each particle is stored every hour. A jupyter-notebook example of the code is shown in fig. 19.

```
[14]: def DeleteParticle(particle, fieldset, time):
          particle.delete()
```

2.4. Particle Set:

```
[12]: pset_v2 = ParticleSet(fieldset=fieldset_total, pclass=JITParticle, lon=lons, lat=lats, time=timep, repeatdt=repeatdt) #, depth=depp
```

3. Executing the simulation:

3.1. Defining parameters

```
[5]: file_name = "test03_v02_ATL_Sep2011_cont1h_MESH.nc"

[15]: output_file = pset_v2.ParticleFile(name=file_name, outputdt=timedelta(hours=1))
```

3.2. Running

```
[16]: pset_v2.execute(pset_v2.Kernel(AdvectionRK4), runtime=timedelta(days=14.), dt=timedelta(minutes=10),
                   recovery={ErrorCode.ErrorOutOfBounds: DeleteParticle}, output_file=output_file)

      start_time = timep.isoformat()
      time_unit_out= "seconds since " + str(pset_v2.time_origin)
      end_time = (num2date(output_file.lasttime_written, time_unit_out)).isoformat()

      add_WOC_nc_attbts(output_file, start_time, end_time, -75.0, 20., 48., 80.
              , input_filename="WOC-L4-CUReul-ENATL-1H: runconv15m_enatl2_d*.nc'"
              , ntitle="North Atlantic 2D horizontal drift of Oil spill for ESA WOC project"
              , summary="This dataset contains the positions of virtual particles released hourly from 10/09/2011 to 25/09/2011 at 15m representin
              , inid = "NATL_OIL_SPILL_DRIFT_1H_CONT"
              , ndepth = "15")

      output_file.export()  # exports the trajectory data to a netcdf file
```

Figure 19. Example of kernel set-up for the execution of the oil spill simulation for scenario 1 (Golden Trader).

3.4. Step 3 : Executing the simulation

3.4.1. Case 1 : Golden Trader oil spill

Once the hydrodynamic input data and particle behaviour kernels are ready, the simulations can be run and saved as netCDFs. To do so, a set of parameters need to be decided:

- **Timestep (dt)** : 10 minutes
- **Repeat dt :** 1 hour throughout the simulation
- **Output timestep (dt) :** Hourly
- **Simulation length:** 14 days
- **Initial positions and release dates :** Positions of released particles (black dots) are shown on fig. 20., at the location of the oil spill event (red dot) on 10/09/2011.

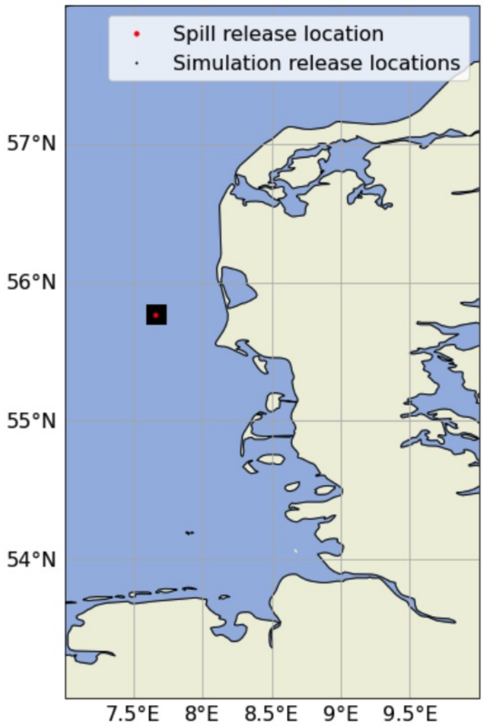


Figure 20. Release locations of the virtual particles representing the Location of the Golden Trader oil spill which took place on the 10/09/2011 off the western coast of Denmark. The red dot shows the exact position where the oil spill took place, and the black dots are the release locations of the virtual particles around the red point. .

3.4.2. Case 2 : Sanchi oil spill

Once the hydrodynamic input data and particle behaviour kernels are ready, the simulations can be run and saved as netCDFs. To do so, a set of parameters need to be decided:

- **Timestep (dt)** : 10 minutes
- **Frequency of re-release of particles (repeat dt) :** 1 hour until the 23rd of January 2018 (following ITOPF's specifications)
- **Output timestep (dt) :** Hourly
- **Simulation length:** 16 days.
- **Initial positions and release dates :** Positions of released particles are shown in fig. 21 (black dots), at the location of the oil spill event on 14/01/2018 (red dot).

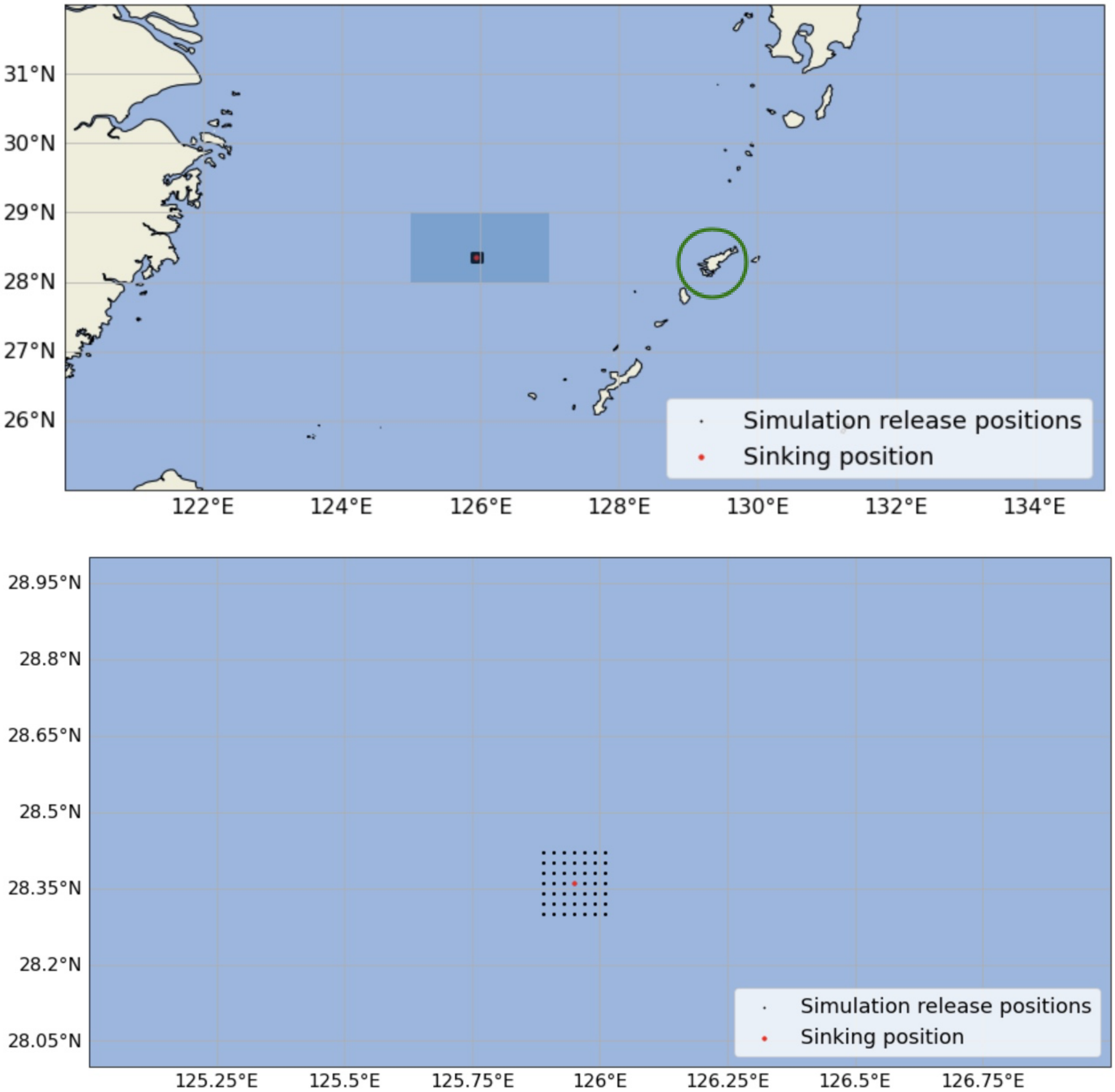


Figure 21. Top: Release locations of the virtual particles representing the Sanchi oil spill which took place on the 14/01/2018 at the East China Sea. Bottom: Zoom in the blue box shown in the top plot. The red dot shows the exact position where the oil spill took place, and the black dots are the release locations of the virtual particles around the red point.

3.5. Product validation

The second version of these oil spill drift simulations has been generated with the second version (v02) of the velocity product. Therefore, like for version 1 of this product, this first simulation with v02 velocities will allow us to identify how and where improvements are needed. The end locations of the oil spill simulations are compared with the information known on the landing/beaching of the oil spills. The simulated trajectories of oil spills are also compared to the CMEMS fields on which the new velocity products are based.

3.5.1. Data

The CMEMS velocity dataset is used to validate the oil spill simulation, see Section 2.5.1 for the details on this dataset.

3.5.1.1. Golden trader oil spill

The collision with a fishing vessel took place off the west coast of Denmark, ~40 km SW of Ringkobing Fjord (see fig. 13). The Swedish coast was impacted, more precisely the Swedish island of Tjörn (Pålsson et al., 2017). For further details see the RB document. Therefore, we compare the end location of the trajectories at both depths (0 and 15m) and for both release types (instantaneous and continuous release) with the expected end location (Denmark and the Swedish island of Tjörn).

In addition to comparing the WOC v02 simulations with the CMEMS simulations at 0m and 15m, the WOC v01 velocity dataset at 0m and 15m is also used. Lastly, we also compare the WOC v02 simulations with simulations done with just the geostrophic component of the WOC v02 velocity dataset at 15m.

3.5.1.2. Sanchi oil spill

To validate the Sanchi oil spill simulations we compare the results with the CMEMS dataset and to information on the beaching/landing of the oil spill. The information on the beaching/landings is described in section 3.2.2.2. Lastly, we also compare the WOC v02 simulations with simulations done with just the geostrophic component of the WOC v02 velocity dataset at 15m.

In this case we cannot compare to the v01 velocity fields, as the data is not available for 2019.

3.5.2. Results

3.5.2.1. Case 1 : Golden trader oil spill

The trajectories simulated are shown in fig. 22. All the particles land/beach on the western Danish coast. Fig. 23 shows the comparison of the simulation done with the other velocity datasets: WOC v02 geostrophic component, WOC v01 at 0m, WOC v01 at 15m, CMEMS at 0m and CMEMS at 15m. All the trajectories are quite similar, except for WOC v01 and CMEMS at 15m, which surprisingly differ with the WOC v02 at 15m. This might be because of the difference in spatial resolution (1/4° and 1/8°).

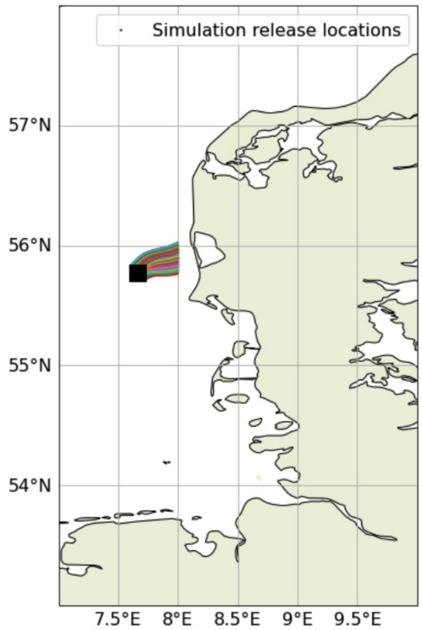


Fig. 22. Oil trajectories after 14 days at 15m depth. Black dots show the starting points (release locations).

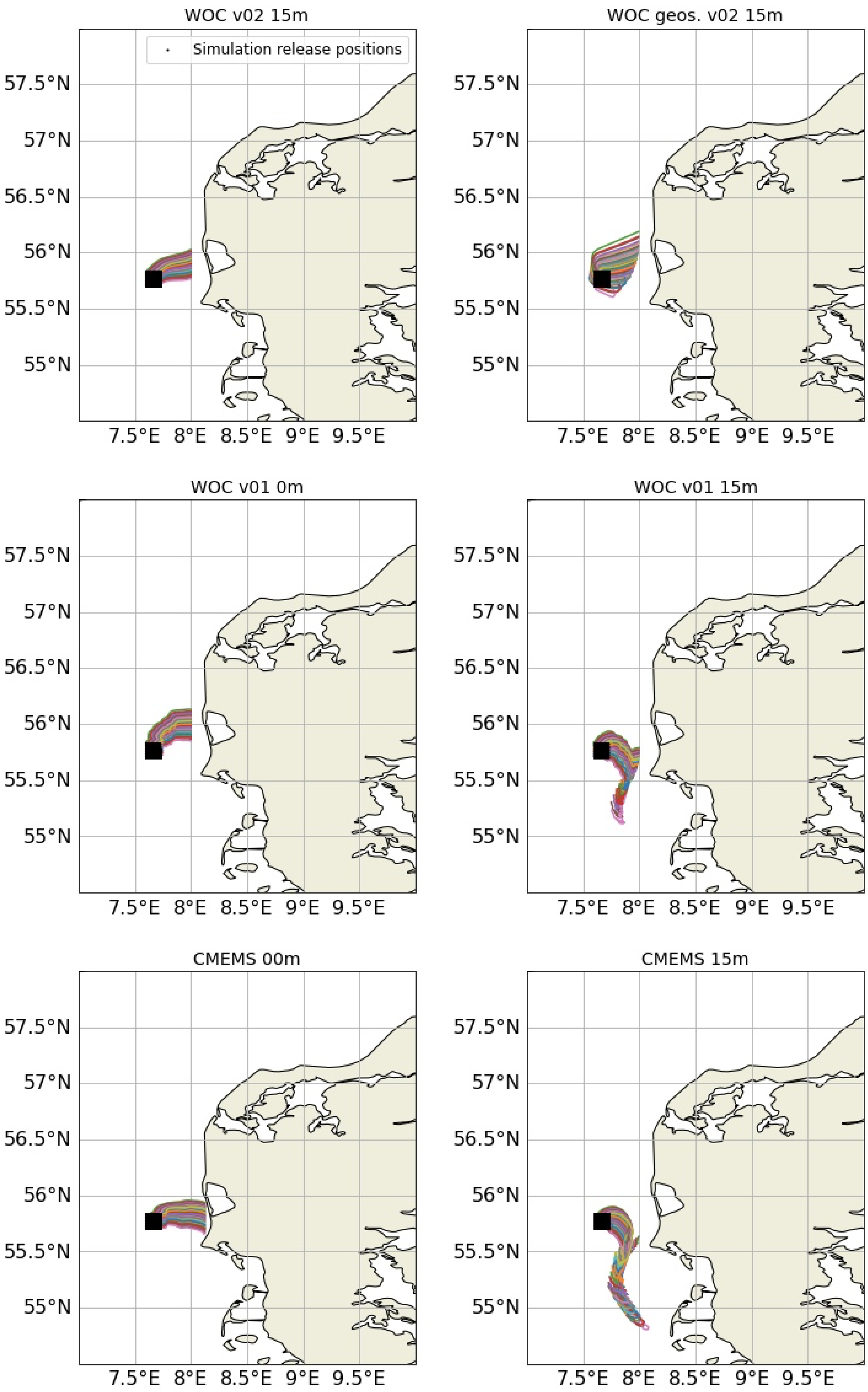


Fig 23. Comparison of the simulated oil spill trajectories with the different velocity dataset. Only the first 50 trajectories are shown.

For both figs. 24 and 25 it can be observed that no matter the velocity dataset (version 1 WOC results not shown), the simulated oil spills end-up on the Danish instead of the Swedish coast. This reflects the importance of using the wind drift, as well as the complexity of simulations close to land.

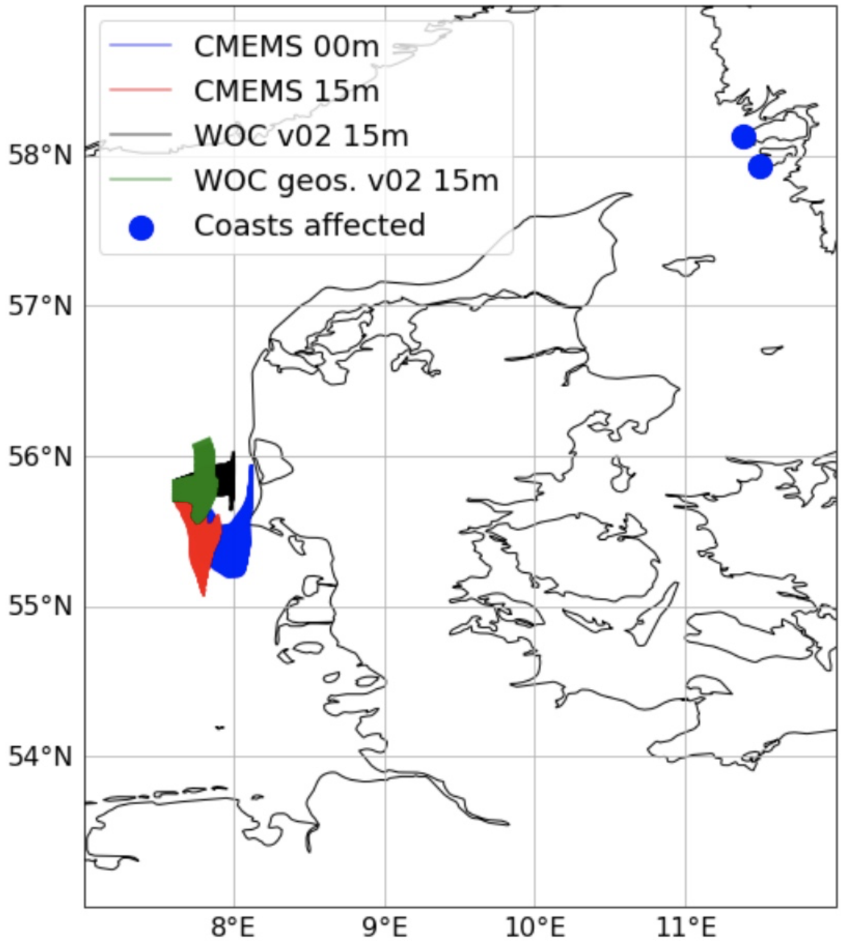


Fig. 24. Simulated oil spill locations on the day of the second landing/beaching on the 16th of September 2011.

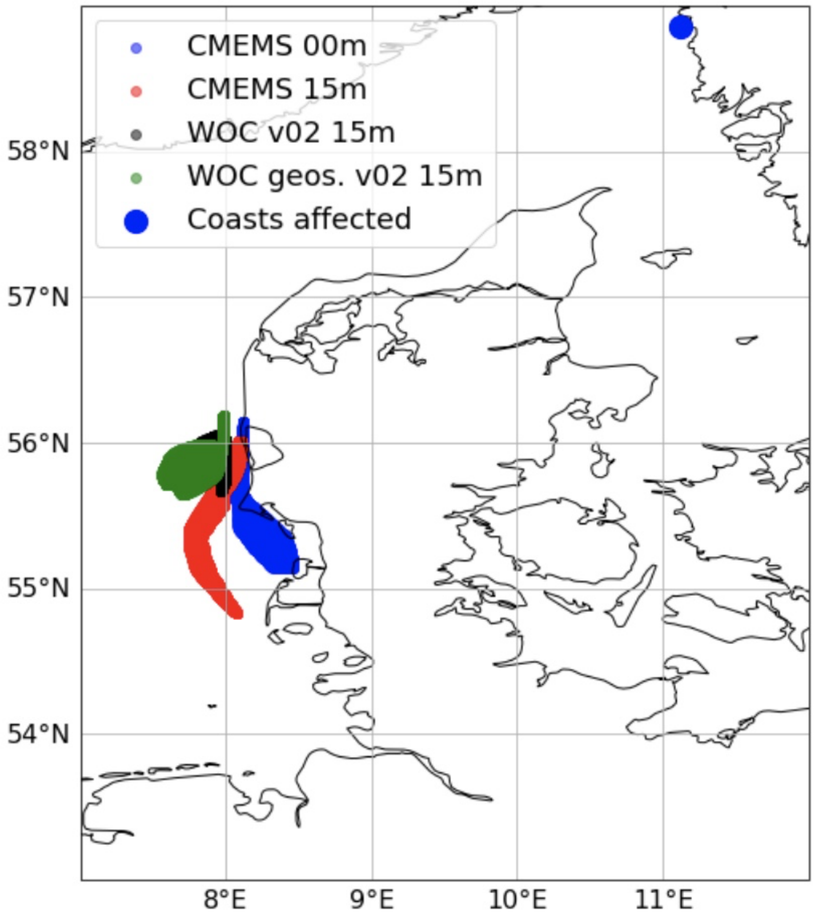


Fig. 25. Simulated oil spill locations on the day of the second landing/beaching on the 21st of September 2011.

Given that the simulation results with just the ocean velocities does not render good results, the quantitative analyses are not shown. Adding the wind effect is necessary to have more realistic simulations.

3.5.2.2. Case 2 : Sanchi oil spill

The trajectories simulated are shown in fig. 26. The displacement of the simulated particles goes east as expected, but farther north than the beaching/landing locations. This is most likely due to the missing effect of the winds that would farther advect the particles east.

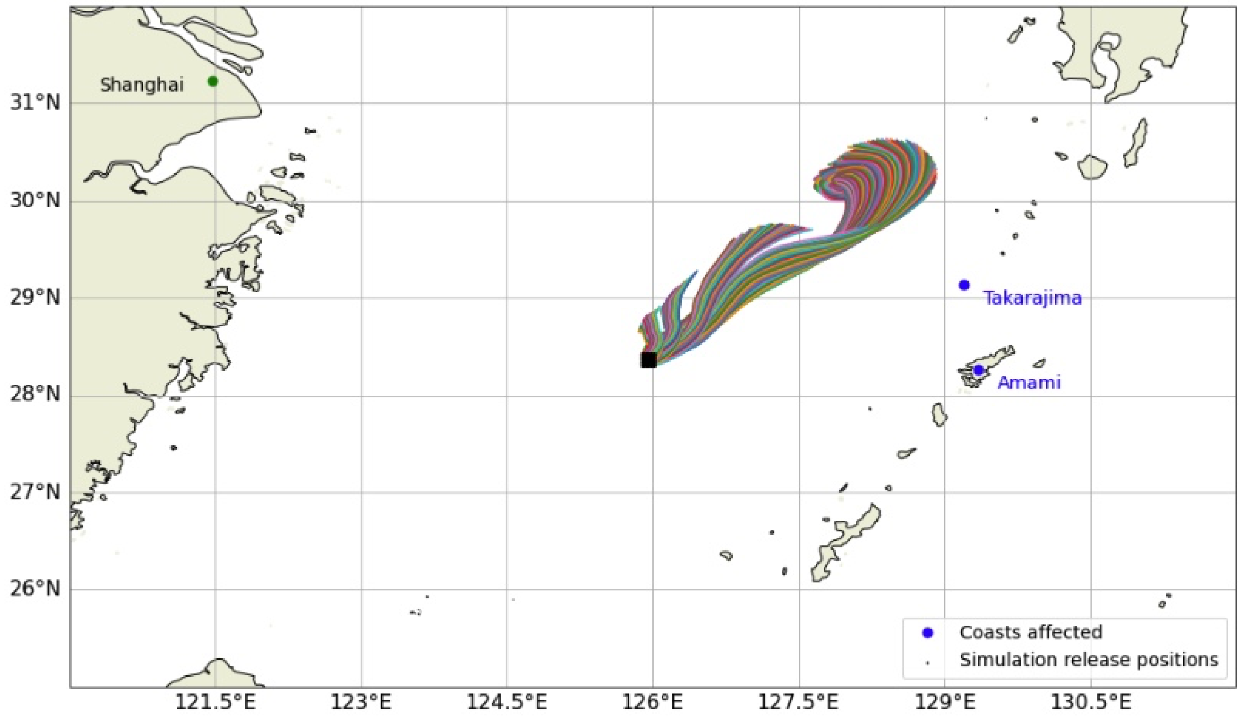


Fig. 26. Sanchi oil spill trajectories after 16 days at 15m depth. Black dots show the starting points (release locations) and blue dots the expected beaching/landing locations.

Figure 27 shows the separation of the particle closest to the beaching areas (Takarajima and Amami Islands) on (or close to) their beaching dates (28/01/2018 and 30/01/2018, respectively). The best results of the arrival of oil spills at the two main landing areas are obtained using the CMEMS 0m velocity data. This is expected as this dataset includes more of the surface dynamics. Then if we compare the WOC v02 datasets with just geostrophy and the total velocity at 15m, the separation is slightly lower with the total velocity. This is a promising first result, but still the surface dynamics is missing (windage, Stokes drift, ...). Also, especially in the Amami case, the CMEMS dataset at 15m provides better results than the WOC dataset.

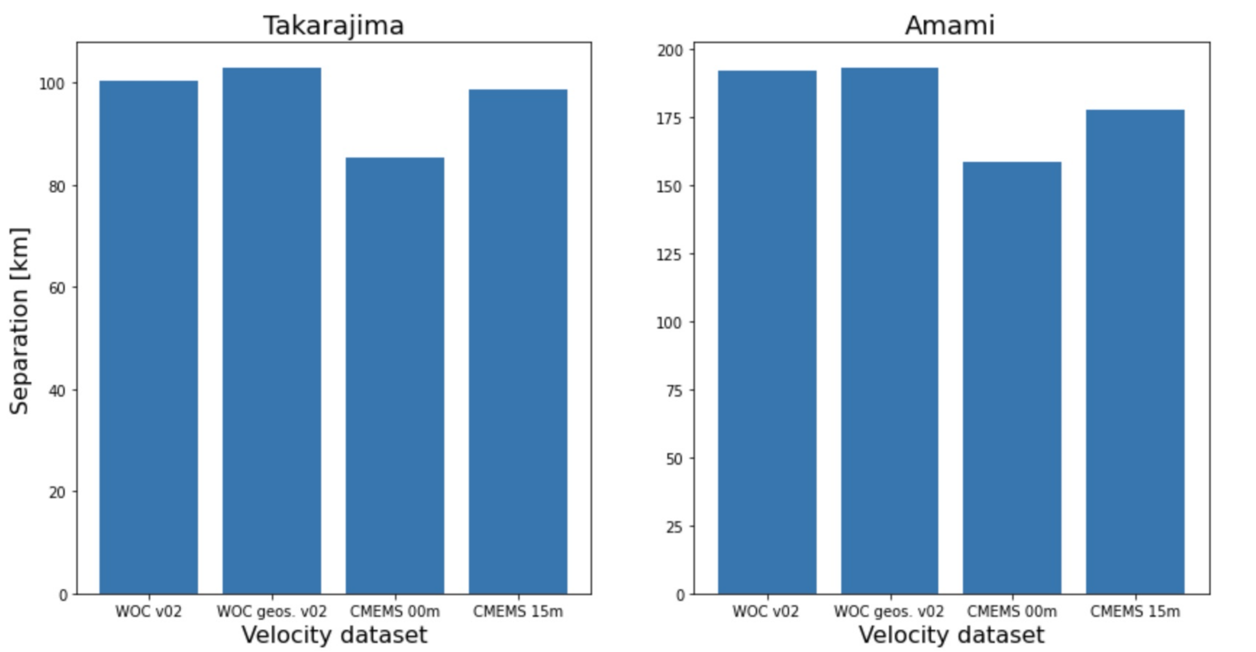


Fig. 27 Distance of closest point to beaching/landing location on day of beaching/landing.