

Développement d'un code numérique de prévision de dérive d'instruments océanographiques.

Présenté par : Guerin Yohan

L3 SVT

Parcours Océanographie Physique et Biogéochimique

Année 2019-2020

Mémoire de stage en Laboratoire de recherche

13 mai 2020

Responsable de stage : Doglioli Andréa

Structure d'accueil : M.I.O

Remerciements

Tout d'abord, je tiens à remercier mon professeur et maître de stage Mr Andrea Doglioli, enseignant-chercheur à l'Institut Méditerranéen d'Océanologie de Marseille, pour le partage de son expérience et sa bienveillance en cette période de pandémie.

Je remercie également tous mes autres professeurs qui m'ont permis d'acquérir les compétences nécessaires à la réalisation de ce stage.

Enfin, je tiens à remercier toutes les personnes qui m'ont conseillé, aidé et relu lors de la rédaction de ce mémoire de stage : ma famille et mes camarades de promotion.

Table des matières

1	Introduction	1
1.1	Plan de déploiement	3
2	Matériels et méthodes	5
2.1	Logiciel utilisé	5
2.2	Données	5
2.3	Modèle numérique	5
2.3.1	Fenêtre spatio-temporelle	5
2.3.2	Approximations physiques	6
2.3.3	Approximations numériques	6
3	Résultats	8
3.1	Modèle d'advection-diffusion	8
3.2	Dérive lagrangienne dans un champ de vitesses stationnaire	9
3.3	Dérive lagrangienne dans un champ de vitesses variable	10
3.4	Sensibilité des paramètres de calcul	11
3.4.1	Dans un champ de vitesses stationnaire AVISO	11
3.4.2	Dans un champ de vitesses variable AVISO	12
3.5	Comparaison des données AVISO et MERCATOR	13
3.6	Comparaison des trajectoires des données AVISO.	14
4	Discussion	15
4.1	Dérive Lagrangienne dans un champ de vitesses variable	15
4.2	Comparaison des trajectoires AVISO et MERCATOR	15
5	Conclusion et perspectives	16
	Références	17

1 Introduction

Dans le contexte actuel de changement climatique, il est essentiel de comprendre les mécanismes de régulation du taux de CO_2 dans l'atmosphère. Un des principaux puits de carbone sur la planète est la pompe biologique de l'océan. Cette dernière est responsable du transfert du carbone organique (CO) vers les profondeurs océaniques. C'est-à-dire vers les sédiments. Cette pompe biologique entraîne du CO sous trois formes. Sous forme de neige marine, de CO dissous et par le biais de migration verticale du zooplancton. Ces flux varient en fonction de la saison et des régions. Le problème aujourd'hui est le peu de connaissance sur le taux de dégradation des différents types de CO. De ce fait, nous ne pouvons pas mesurer correctement les flux entre la surface et la zone mésopélagique. Le puits de carbone que constitue l'océan profond reste encore mal évalué. Des chercheurs du laboratoire MIO¹ avaient décidé de programmer une mission océanographique au large de Toulon mi-Avril. Cependant, suite à la pandémie du SARS-CoV-19, elle est en attente de reprogrammation. Cette mission s'inscrit dans le cadre de la campagne EMSO ANTARES 2020 LEG2.

D'après le document de pré-mission fourni par le MIO, cette campagne vise à estimer :

- Les proportions relatives liées aux différentes voies d'export gravitationnel ainsi que l'export de CO total à travers différents niveaux de la zone mésopélagique ;
- la caractérisation chimique (CO et biominéraux) de trois types distincts de neige marine (cellules seules, agrégats, déjections de zooplancton) en relation avec leurs tailles respectives ;
- les taux de dégradation associés à ces trois types de particules.

Afin de mener à bien la mission, l'équipe va déployer 3 méthodes de prélèvement d'eau (Figure 1).

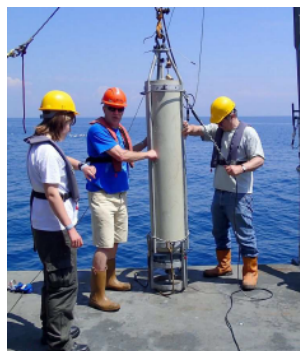
- des bouteilles Niskin montées sur rosettes (a) ;
- des marines snow catchers (b) ;
- de pièges à sédiments (c).

1. Mediterranean Institute of Oceanography

Ces instruments vont être montés sur une ligne dérivante (Figure 1d).



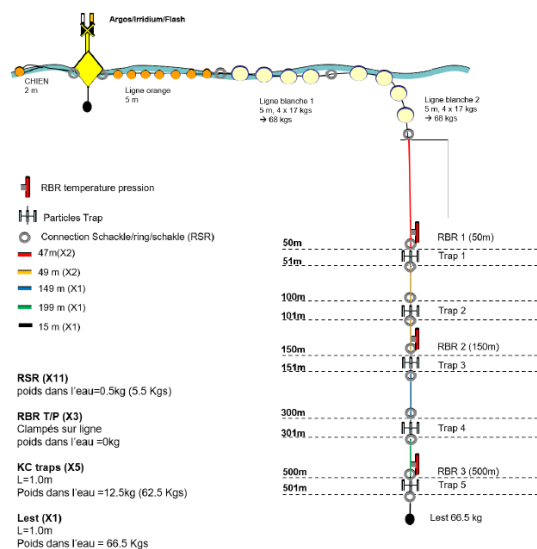
(a) Bouteilles Niskin montées sur rosette
© IRD RUITTON Sandrine



(b) Marine snow catcher
© rapport de pré-mission EMSO ANTARES 2020 LEG2



(c) piège à sédiments
© <https://www.bionef.fr/produit/piege-a-sediments-type-saarso/>

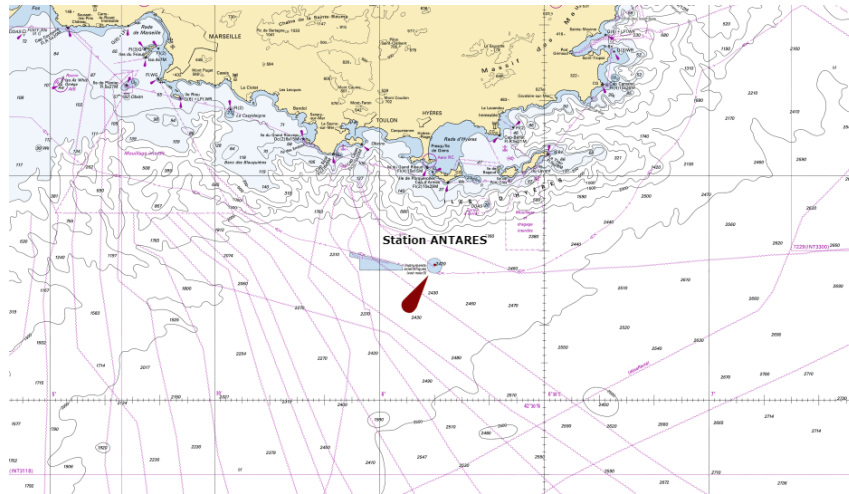


(d) Ligne dérivante
© rapport de pré-mission EMSO ANTARES 2020 LEG2

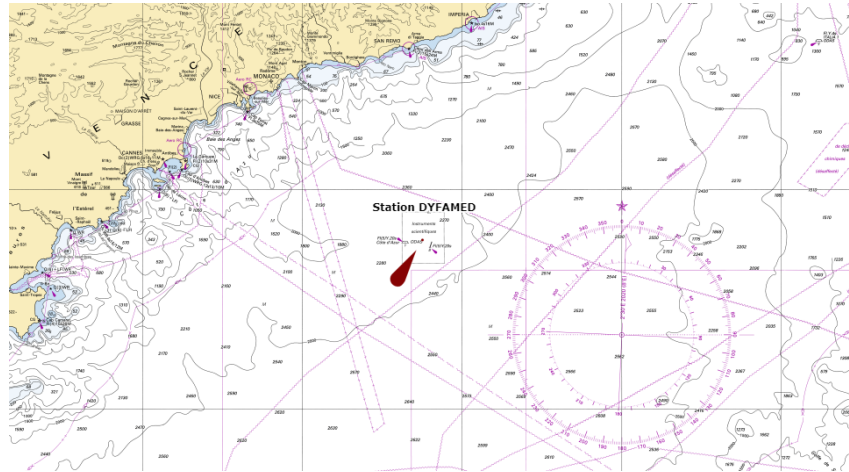
FIGURE 1: Instruments de prélèvements montés sur la ligne dérivante.

1.1 Plan de déploiement

Les chercheurs du laboratoire du MIO ont décidé de larguer les lignes dérivantes (Figure 1d) autour du site ANTARES² aux coordonnées $42^{\circ}48'N$; $6^{\circ}10'E$ (Figure 2a) ou DYFAMED³ aux coordonnées $43^{\circ}42'N$; $7^{\circ}90'E$ (Figure 2b). Le but est d'avoir un suivi du taux de sédimentation de neige marine sur 4 à 5 jours, pour le mettre en relief avec les relevés à long terme de la station ANTARES ou DYFAMED. Le choix du site se fera en fonction des autorisations de l'administration de la Marine Militaire de Toulon et de la météo la semaine de mission.



(a) Station ANTARES (au large de Toulon)



(b) Station DYFAMED (au large de Monaco)

FIGURE 2: Zones d'étude ; Échelle : 1/866 688 ; Tiré de <https://data.shom.fr/>

2. Astronomy with a Neutrino Telescope and Abyss Environmental Research
3. Dynamics of Atmospheric Fluxes in the Mediterranean sea

Dans cette zone se trouve le courant Liguro-provençal aussi appelé Courant Nord. Ce courant est une veine à environ 100 m de profondeur d'une vitesse moyenne de 0.6 m.s^{-1} [Meloni et al., 2019]. Il résulte de la rencontre des eaux superficielles d'Atlantique passant à l'Ouest de la Corse (MAW⁴) et du courant venant du bassin orientale de la méditerranée passant à l'Est de la Corse. Ces deux veines se rejoignent le long des côtes Ligures puis dérivent vers l'Ouest grâce à l'effet géostrophique. Puis, passant successivement au large de Monaco et de Toulon, il continue vers les côtes espagnoles [Milot and Taupier-Letage, 2005].

A l'aide d'images satellites et de modèles numériques du courant liguro-provençal, le travail de ce stage consiste à simuler les trajectoires sur environ une semaine des lignes dérivantes déployées pour la mission. La méthode est d'appliquer un modèle de dérive Lagangienne à des particules représentant les bouées déployées. C'est à dire à suivre le déplacement des particules en calculant leur trajectoire à l'aide d'un modèle auto-régressif en utilisant des variables pseudo-aléatoires afin de simuler une dérive naturelle e.g [Messié et al., 2020].

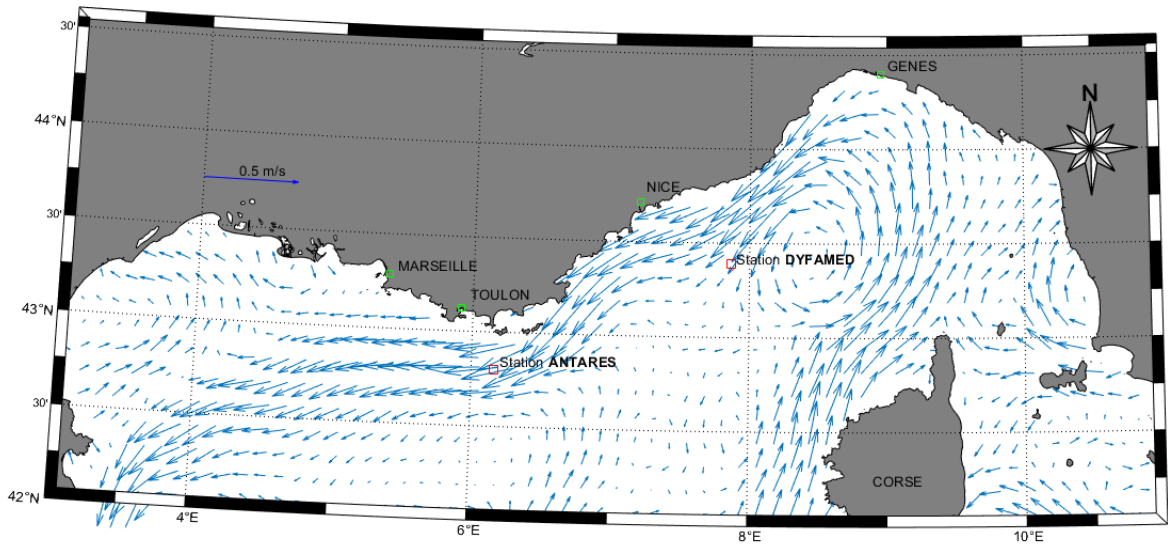


FIGURE 3: Situation de surface en méditerranée Occidentale

Station ANTARES : $42^{\circ}48'N$; $6^{\circ}10'E$; Station DYFAMED : $43^{\circ}42'N$; $7^{\circ}90'E$.

4. Modified Atlantique Water, ou "eaux de surface atlantiques modifiées"

2 Matériels et méthodes

2.1 Logiciel utilisé

Afin de répondre à la problématique, nous avons utilisé le logiciel Matlab dans le but de faire un algorithme. Le logiciel utilise des bibliothèques de fonction préprogrammées afin de faciliter le script et d'avoir un rendu de cartes et de graphiques plus explicites. C'est notamment le cas des bibliothèques *netcdf*⁵ pour lire les données satellites issues de **Marine Copernicus** et *m_map*⁶ pour faire les cartes. La version Matlab utilisée dans ces travaux est R2020a.

2.2 Données

Marine Copernicus ou CMEMS⁷ est un programme européen qui permet de surveiller les paramètres environnementaux de la Terre. Il possède 6 services afin de surveiller l'entière de la Terre. Nous allons cependant nous intéresser seulement aux données marines. Ces données peuvent être mesurées de manière *in situ* ou par satellites (données **AVISO**). Elles peuvent aussi venir de modèles numériques (données MERCATOR à **haute** et **basse** résolution spatiale). Depuis 2014 le programme a lancé 6 satellites. Cela nous permet aujourd'hui d'avoir des observations sur 6 ans. Ainsi nous avons pu extraire les vecteurs vitesse « u et v » en fonction de la latitude et de la longitude. Cela afin de simuler la dérive de particules Lagrangiennes dans le courant liguro-provençal.

2.3 Modèle numérique

2.3.1 Fenêtre spatio-temporelle

Afin de modéliser la trajectoire de points numériques nous avons besoin de plusieurs paramètres tels que les valeurs du champ des vitesses sur 7 jours. De plus, nous devons définir une aire d'étude. L'aire d'étude a été choisie afin de pouvoir localiser la zone A (au large de Toulon) et la zone B (au large de Monaco). La résolution spatiale des données est de 0.125° et la résolution temporelle de 1 jour, nous avons pris comme fenêtre d'étude : Longitude : 35°E à 11°E. Latitude : 41°N à 44.5°N. Période temporelle : du 14/04/2019 au 20/04/19 (7 jours).

Nous savons que dans cette zone se trouve le courant Liguro-Provençal (Figure 3). De ce fait, nous pouvons aisément prévoir la trajectoire des bouées larguées.

5. Unidata, (2020) : Network Common Data Form (NetCDF) version 4.7.4 [Matlab]. Boulder, CO : UCAR/Unidata. (<https://doi.org/10.5065/D6H70CW6>).

6. Pawlowicz, R., 2020. "*m_map* : Un package de création de carte pour MATLAB", version 1.4m, [Windows 10], accessible en ligne à www.eoas.ubc.ca/~rich/map.html.

7. Copernicus Marine Environment Monitoring Service.

2.3.2 Approximations physiques

Nous considérons ici que chaque particule modélisée correspond à une bouée. Nous considérons aussi que les champs de vitesses issus du CMEMS sont des champs de vitesses moyennes journalières. De plus, nous considérons la bouée comme une particule se situant à la surface de l'eau où la vitesse de déplacement est uniquement due aux forces appliquées sur celle-ci. Enfin, nous négligeons les vitesses verticales. Nous modélisons donc notre déplacement dans un plan Cartésien en deux dimensions, seulement suivant x et y .

De ce fait, nous pouvons déterminer les positions successives des points dans un espace en deux dimensions en fonction du temps à partir de :

$$\vec{v} = \frac{d\vec{r}}{dt} \text{ avec } \vec{r} = (x, y) \text{ et } \vec{v} = (u, v) \quad (1)$$

On considère le fluide étudié comme écoulement turbulent et non laminaire. De ce fait, afin de modéliser une dérive Lagrangienne nous utilisons la décomposition des vitesses de Reynolds $u = \bar{u} + u'$ et $v = \bar{v} + v'$. Ceci nous permet de décomposer nos vitesses u et v en une composante advective \bar{u} et \bar{v} notre vitesse moyenne et une composante diffusive u' et v' représentant un faible déplacement non prévisible de notre particule.

2.3.3 Approximations numériques

Nous savons que :

$$\vec{v} \simeq \frac{\vec{r}^{t+\Delta t} - \vec{r}^t}{\Delta t} \Rightarrow \vec{r}^{t+\Delta t} = \vec{r}^t + \vec{v}\Delta t \quad (2)$$

De ce fait, en discrétisant la formule (2) avec $\vec{r} = (x, y)$ et $\vec{v} = (u, v)$ on a :

$$x^{t+\Delta t} = x^t + u\Delta t \text{ et } y^{t+\Delta t} = y^t + v\Delta t \quad (3)$$

Avec la décomposition de Reynolds $u = \bar{u} + u'$ et $v = \bar{v} + v'$:

$$x^{t+\Delta t} = x^t + (\bar{u} + u')\Delta t \text{ et } y^{t+\Delta t} = y^t + (\bar{v} + v')\Delta t \quad (4)$$

On a donc \bar{u} et \bar{v} la moyenne du champ des vitesses sur une période t issue des données de Marine Copernicus. Cependant nous devons numériser u' et v' , les écarts par rapport à la moyenne appelée composante diffusive. Ces composantes diffusives sont des variables pseudo-aléatoires générées à l'aide d'une fonction préprogrammée dans l'interface de Matlab : *randn*. La fonction *randn* est contrôlée par les paramètres statistiques d'une loi Normale. C'est-à-dire que lorsque l'on tire aléatoirement une succession à l'aide de cette fonction, nous obtenons une distribution Gaussienne. Suivant ainsi une loi normale de paramètre $N(0,1)$.

On pose :

$$u' = \mu_x \quad \text{et} \quad v' = \mu_y \quad (5)$$

Finalement :

$$x^{t+\Delta t} = x^t + (\bar{u} + \mu_x)\Delta t \quad \text{et} \quad y^{t+\Delta t} = y^t + (\bar{v} + \mu_y)\Delta t \quad (6)$$

Au regard des données AVISO du CMEMS on a $\bar{u} < u'$. De ce fait, il faut pondérer u' avec \bar{u} . Nous devons donc contrôler la distribution de μ_x et μ_y afin qu'elle prenne une valeur réaliste pour une composition turbulente. Nous déterminons ainsi un facteur d'étalement appelé C . Nous allons de ce fait multiplier notre composante diffusive μ_x et μ_y par un coefficient C .

μ_x devient $C\mu_x$ et μ_y devient $C\mu_y$. On sait que le courant est d'environ $6.10^{-1} \text{ m.s}^{-1}$ et qu'il varie de $\pm 6.10^{-3} \text{ m.s}^{-1}$. Dans notre cas nous prendrons alors $C = 10^{-3}$.

$$x^{t+\Delta t} = x^t + (\bar{u} + \mu_x.10^{-3})\Delta t \quad \text{et} \quad y^{t+\Delta t} = y^t + (\bar{v} + \mu_y.10^{-3})\Delta t \quad (7)$$

On a ainsi notre modèle auto-régressif à implémenter dans Matlab afin de simuler la trajectoire des bouées.

Afin d'optimiser l'algorithme nous avons fait des tests de sensibilité des paramètres. Le but étant de choisir des temps d'intégration différents afin de comparer les trajectoires de bouées. Cela doit se faire sans la partie diffusive du modèle afin de mesurer uniquement l'impact du temps d'intégration. En faisant cela, nous pouvons ainsi juger le pas de temps optimal pour le calcul tout en ayant des résultats cohérents [Qiu et al., 2011].

Une fois que l'algorithme avec les données AVISO est validé, une comparaison avec les données MERCATOR est nécessaire. En effet, les champs de vitesse AVISO sont issus de relevés en surface alors que les champs de vitesse MERCATOR sont issus de modèles à 15 m de profondeur. Nous estimons ici que la force appliquée sur les bouées est maximale à 15 m.

De ce fait, nous avons réalisé avec le même algorithme, une comparaison des trajectoires avec les données AVISO et MERCATOR.

3 Résultats

3.1 Modèle d'advection-diffusion

La première étape de la modélisation numérique fut de faire un algorithme basé sur un modèle d'advection-diffusion (Annexe I). On a alors la trajectoire de différentes particules sur un temps donné dans un référentiel cartésien. La fiabilité du programme est vérifiée en prenant un champ de vitesses constant. Afin de rendre le programme plus rapide, le pas de temps des calculs est de l'ordre de la minute et non de la seconde. Cependant, le risque d'augmenter le temps d'intégration est de rendre le modèle trop instable. Afin de valider le modèle, une série de tests a été fait avec succès sur celui-ci.

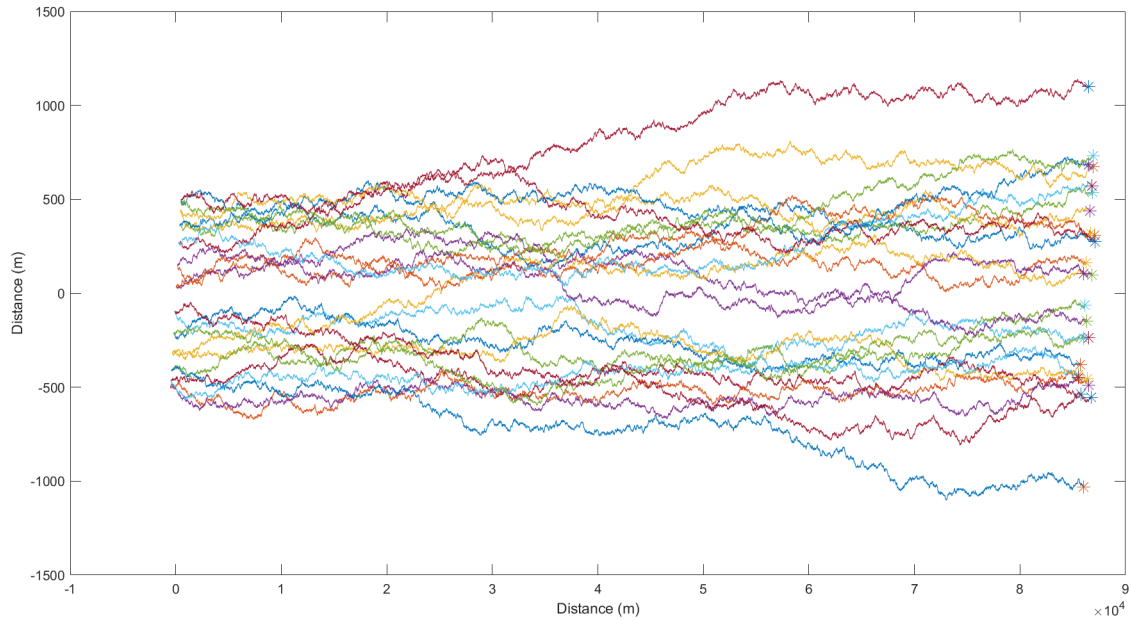


FIGURE 4: Modèle d'advection-diffusion dans un référentiel Cartésien. $u = 1 \text{ m.s}^{-1}$; $v = 0 \text{ m.s}^{-1}$; $t = 86\,400 \text{ s}$; $n = 25$ particules

3.2 Dérive lagrangienne dans un champ de vitesses stationnaire

L'étape suivante est d'insérer dans l'algorithme le champ des vitesses issu des données AVISO de Marine Copernicus (Annexe II). De plus, il faut transformer le référentiel cartésien où $\vec{r}=(x,y)$ en un référentiel géographique où $\vec{l}=(\text{longitude},\text{latitude})$. C'est là que la librairie «*m_map*» est utilisée. On a donc un programme qui simule les trajectoires de particules dans un champ de vitesse stationnaire. On peut ainsi simuler les trajectoires sur un à plusieurs jours. Comme cité précédemment, la résolution temporelle des données est de 1 jour et d'une résolution spatiale de 0.125 degré. Le problème de cet algorithme est qu'il ne prend pas en compte la variation du champ de vitesses toutes les 24h (temps de mise à jour des données satellite). Il faut donc modifier l'algorithme afin de changer le champ des vitesses toutes les 24h.

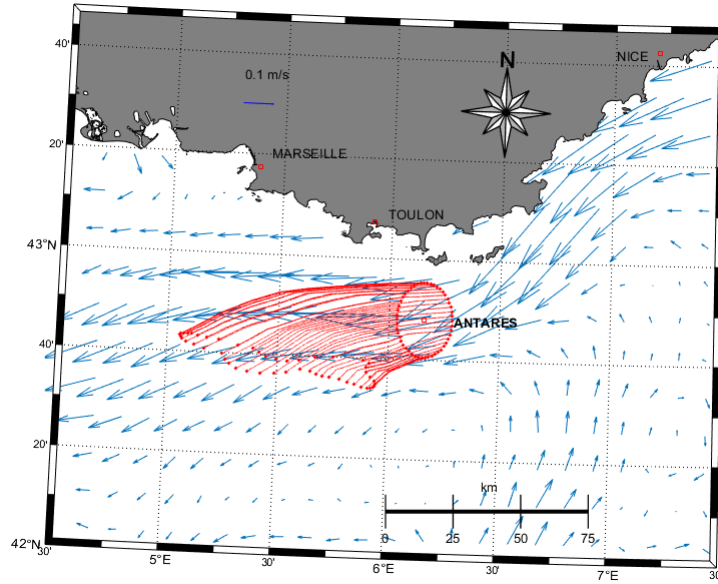
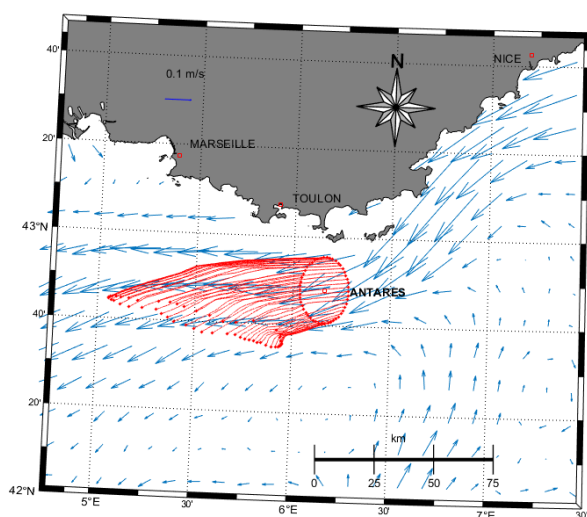


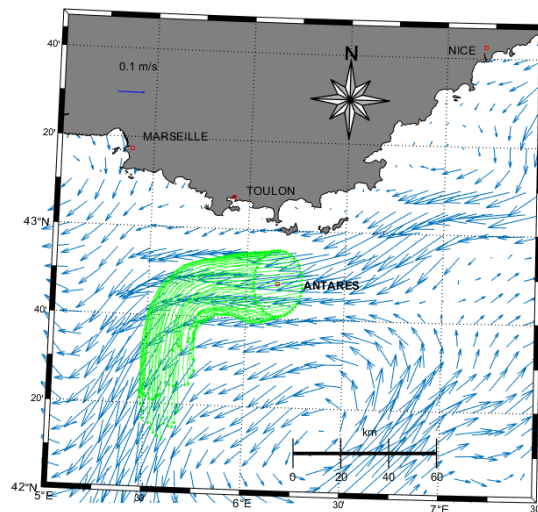
FIGURE 5: Simulation de dérive Lagrangienne de 50 particules dans un champ de vitesses stationnaire ; Dates : 14/04/2019 au 18/04/2019 ; Départ selon un cercle de rayon 0.125° centré sur $6.16^\circ E$; $42.8^\circ N$

3.3 Dérive lagrangienne dans un champ de vitesses variable

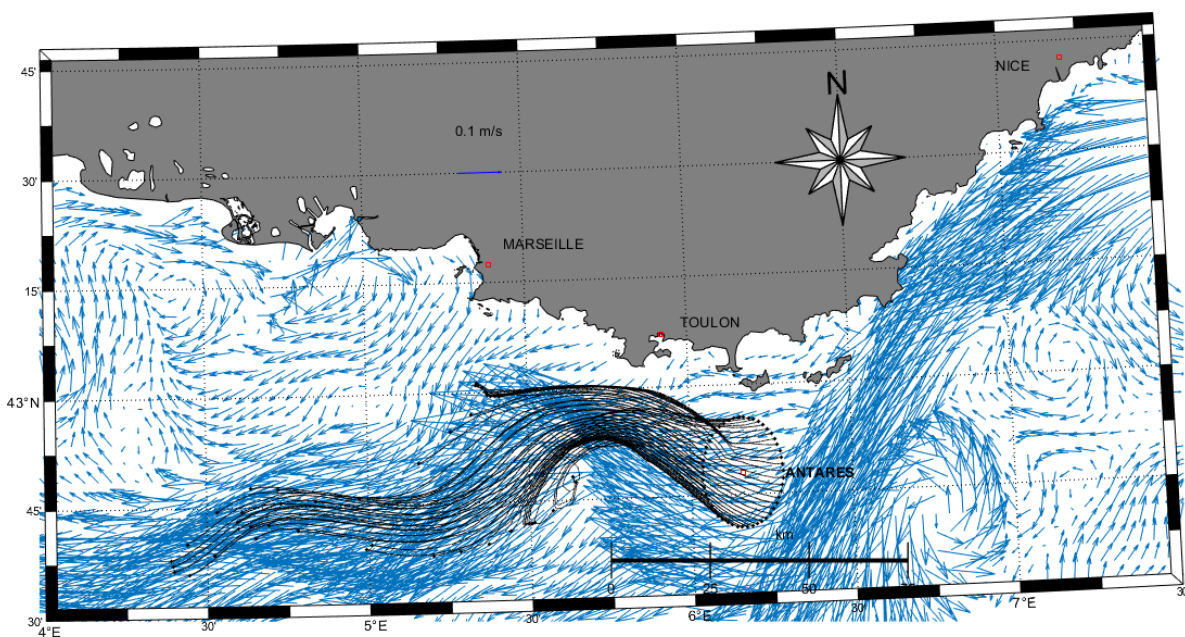
En prenant les mêmes paramètres que précédemment mais en modifiant le champ toutes les 24 h (Annexe III) et en l'appliquant aux champs de vitesses AVISO (Figure 6a) et au champ MERCATOR haute (Figure 6b) et basse résolution spatiale (Figure 6c) on a :



(a) Trajectoires avec les données AVISO



(b) Trajectoires avec les données MERCATOR à basse résolution



(c) Trajectoires avec les données MERCATOR à haute résolution

FIGURE 6: Simulation de dérive Lagrangienne de 50 particules dans un champ de vitesse variable ; Dates : 14/04/2019 au 18/04/2019 ; Départ selon un cercle de rayon 0.125° centré sur $6.16^\circ E$; $42.8^\circ N$

3.4 Sensibilité des paramètres de calcul

Le test de sensibilité se fait sans la partie diffusion. Cela nous permet de mesurer uniquement l'impact du temps d'intégration sur la simulation sans l'effet diffusion.

3.4.1 Dans un champ de vitesses stationnaire AVISO

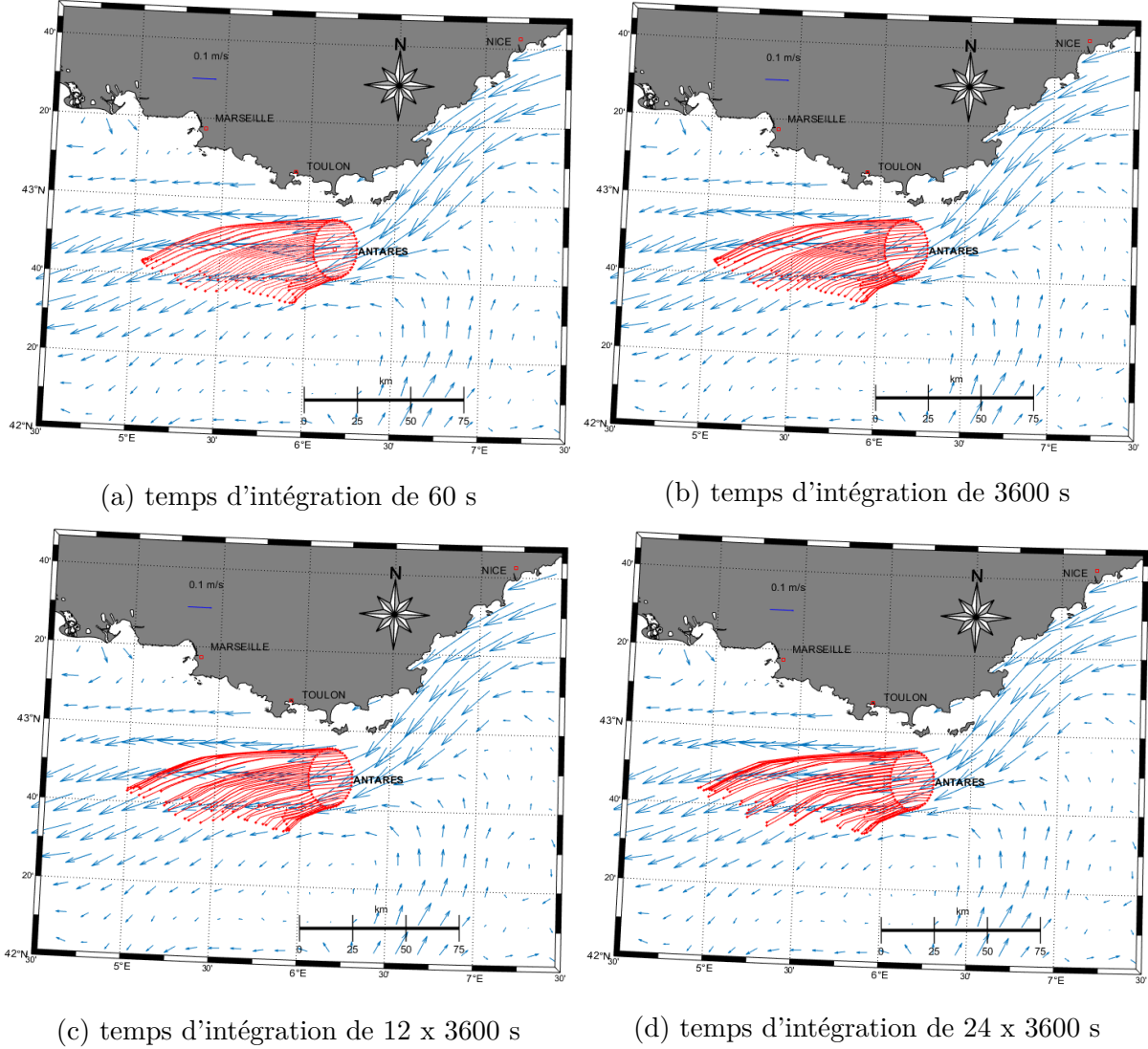


FIGURE 7: Simulation de dérive Lagrangienne de 50 particules dans un champ de vitesse stationnaire à différents pas de temps ; Dates : 14/04/2019 au 18/04/2019 ; Départ selon un cercle de rayon 0.125° centré sur $6.16^\circ E$; $42.8^\circ N$

3.4.2 Dans un champ de vitesses variable AVISO

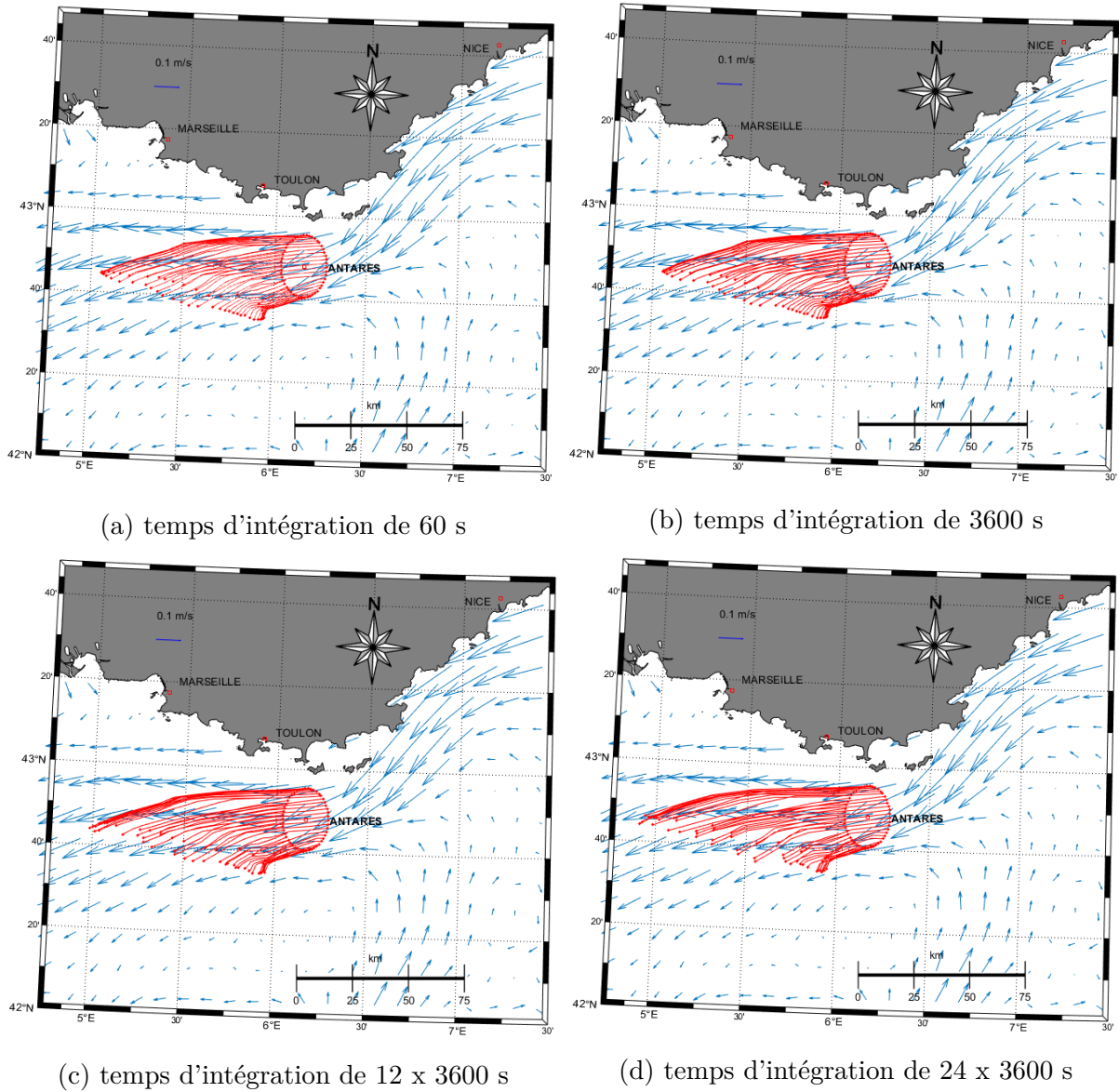


FIGURE 8: Simulation de dérive Lagrangienne de 50 particules dans un champ de vitesse variable à différents temps d'intégration ; Dates : 14/04/2019 au 18/04/2019 ; Départ selon un cercle de rayon 0.125° centré sur $6.16^\circ E$; $42.8^\circ N$

On observe que pour des temps d'intégration de 60 s à 3600 s, les trajectoires divergent très peu. C'est seulement pour des temps d'intégration de 12×3600 s que les trajectoires varient significativement. La distance parcourue par les particules est beaucoup plus grande. Cela est dû au fort gradient de vitesses. On observe ici un courant fortement cisailé. Il peut donc être utile d'utiliser un temps d'intégration de 100 s afin de gagner en temps de calcul tout en gardant des trajectoires cohérentes.

3.5 Comparaison des données AVISO et MERCATOR

Ici nous avons la comparaison des simulations de trajectoires issues des données AVISO en rouge et des données MERCATOR haute résolution spatiale en noir. La dérive se fait sur 5 jours, entre le 14/04/2019 et le 18/04/2019 pour un pas de temps de 100 s (Annexe IV).

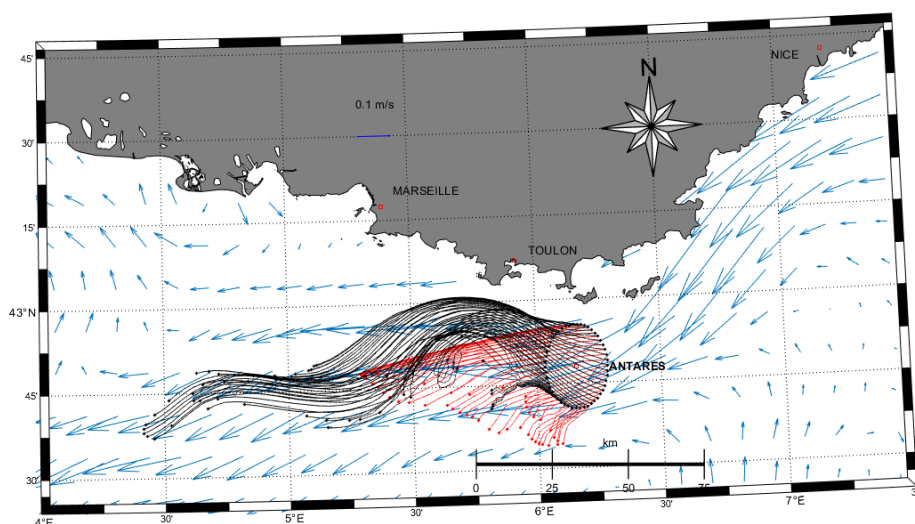


FIGURE 9: Superposition des simulations de dérive Lagrangienne de 50 particules dans un champ de vitesse variable avec les données AVISO (rouge) et MERCATOR (noir); Dates : 14/04/2019 au 18/04/2019; Départ selon un cercle de rayon 0.125° centré sur $6.16^\circ E$; $42.8^\circ N$

On observe des trajectoires très différentes. Nos particules sont dispersées avec des trajectoires courtes pour les données AVISO alors que pour les données MERCATOR, les trajectoires sont plus longues et les particules moins dispersées.

3.6 Comparaison des trajectoires des données AVISO.

Nous observons à quelques différences près un même schéma de déplacement des particules. En partant de la station ANTARES les particules qui se trouvent plus au Nord ont un trajet plus long que celles se situant au Sud du point de départ.

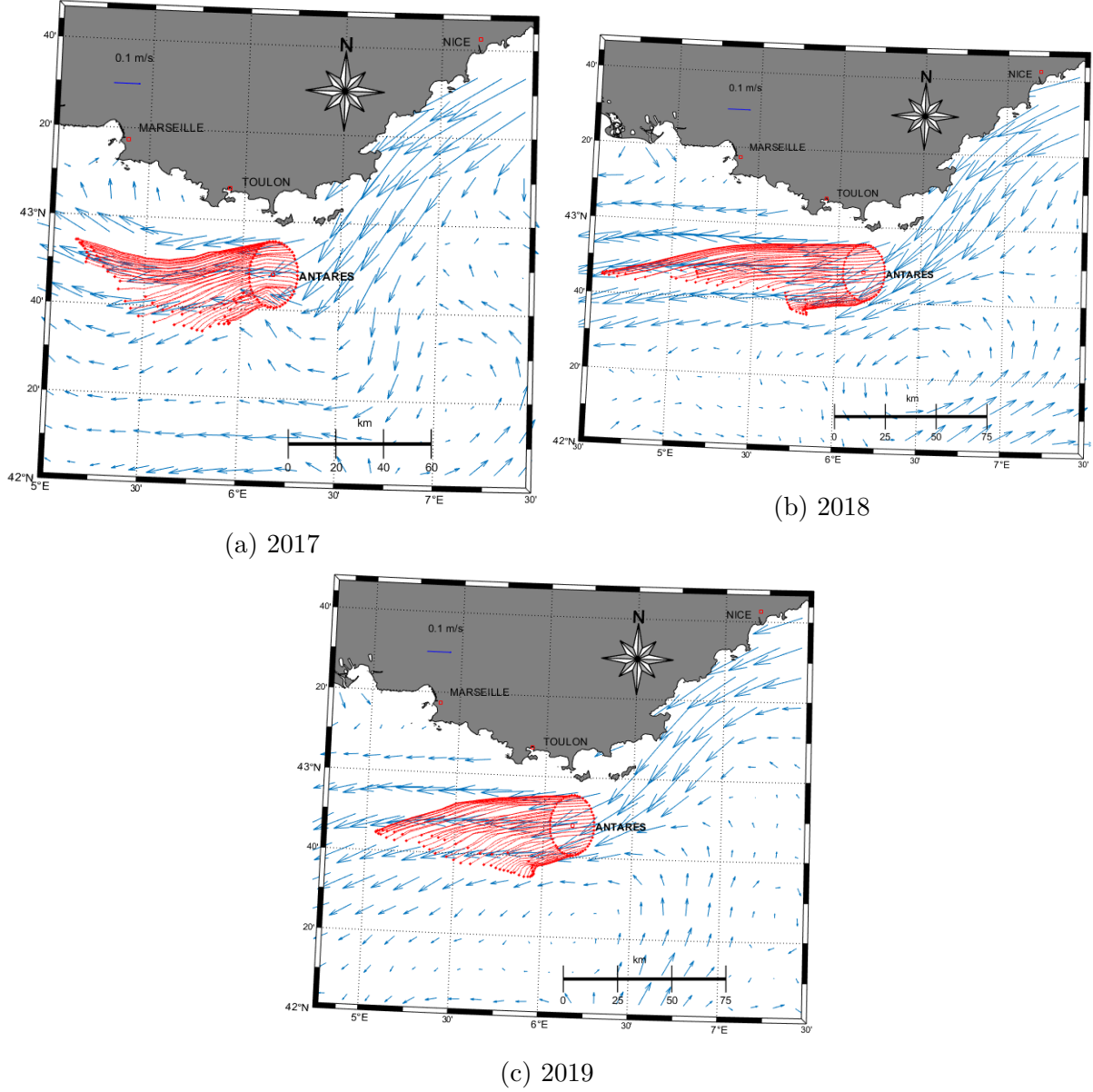


FIGURE 10: Simulations de dérive Lagrangienne de 50 particules dans un champ de vitesse variables du 14/04 et 18/04 en 2017, 2018 et 2019 ; Départ selon un cercle de rayon 0.125° centré sur $6.16^\circ E$; $42.8^\circ N$

4 Discussion

4.1 Dérive Lagrangienne dans un champ de vitesses variable

On observe (Figure 6) que suivant les données, les trajectoires ne sont pas les mêmes. En effet, sans changer l'algorithme mais en choisissant des données AVISO en surface ou MERCATOR à 15 m à différentes résolutions spatiales nous n'avons pas les mêmes trajectoires. La distance de dérive des bouées peut aller du simple au triple. On remarque aussi que les vecteurs de vitesse (bleu) sont moins nombreux et de faible intensités pour la simulation avec les données AVISO. Au contraire, pour les simulations avec les données MERCATOR basse et haute résolution spatiale, les vecteurs sont plus nombreux de plus grande intensité. De plus, on observe une forte disparité entre les deux jeux de données MERCATOR. Ces champs de vitesses sont à la même profondeur mais les trajectoires divergent significativement. Cela s'explique par la différence de résolution spatiale. En augmentant cette dernière, le champ de vitesse devient plus fiable.

Les trajectoires modélisées ici sont sur 5 jours puisque les chercheurs du MIO ont prévu une dérive de maximum 4 à 5 jours. En laissant les bouées dériver moins de 5 jours, cela permettra d'avoir des trajectoires plus courtes et moins dispersées dans l'espace.

On observe aussi que les trajectoires d'une même simulation changent très peu entre un temps d'intégration de 60 s et 3600 s. C'est seulement avec temps d'intégration de 12 h que les trajectoires divergent significativement. Avec un temps d'intégration de 100 s on peut avoir des trajectoires cohérentes tout en optimisant le temps de calcul.

4.2 Comparaison des trajectoires AVISO et MERCATOR

Les trajectoires sur une même période temporelle ne sont pas les même (Figure 6). Cela est probablement dû aux approximations physiques. Effectivement, nous avons pris les champs de vitesses en surface pour les données AVISO alors que pour les données MERCATOR nous avons pris un champ de vitesses à 15 m. De plus, nous observons une intensité du champ des vitesses plus importante dans les données à 15 m (Figure 6b). Un champ de vitesse différent et une vitesse moyenne plus importante engendre de fortes disparités entre les trajectoires des données AVISO et MERCATOR. Il est partiellement correcte de considérer que la trajectoire des bouées résulte uniquement du champ variable à 15 m. En effet, il aurait été judicieux de considérer la ligne de bouée comme une fonction Logarithmique (Figure 1d). De ce fait, nous aurions intégré les forces sur toute la longueur de la ligne dérivante. Pour des raisons de temps et de ressources informatiques nous n'avons pas pu faire cela.

5 Conclusion et perspectives

Nous avons pour chaque jeu de données des trajectoires différentes. Cela est dû à la différence des paramètres de chaque jeu de données. Même s'il est judicieux de choisir la simulation de dérive avec la plus haute résolution spatiale (MERCATOR haute résolution), il est compliqué de prévoir la trajectoire exacte des particules. Il en ressort des simulations de dérive Lagrangienne de 50 particules, qu'en augmentant le temps de dérive, les particules sont plus dispersées. Nous avons ici des simulations de trajectoires les plus pessimistes. Nous avons estimé une dérive sans frottements. Nous avons aussi négligé l'effet "ancrage" de la ligne dérivante due à sa morphologie. Les trajectoires des données MERCATOR nous montrent que les particules suivent un même chemin mais sont moins dispersées. Les trajectoires des données AVISO sont plus éparpillées mais sont plus proches du point de départ. De plus, nous n'avons pas au fil des années des mêmes trajectoires, ce qui accentue les incertitudes des trajectoires simulées. En conséquence, nous ne pouvons à ce stade déterminer la trajectoire exacte des bouées qui seront déployées. Nous pouvons seulement estimer une aire d'arrivée des bouées. Cependant, cette aire sera de plusieurs centaines de kilomètres carrés. Avec ces simulations, nous ne pouvons pas dire avec précision où récupérer les bouées déployées. Afin de simuler une dérive plus précise, nous pourrions modéliser les particules Lagrangiennes en 3 dimensions. Les vitesses seront alors intégrées sur toute la longueur de la ligne dérivante.

Références

- [Meloni et al., 2019] Meloni, M., Bouffard, J., Doglioli, A. M., Petrenko, A., and Valladeau, G. (2019). Toward science-oriented validations of coastal altimetry : Application to the Ligurian Sea. *Remote Sensing of Environment*, 224 :275–288.
- [Messié et al., 2020] Messié, M., Petrenko, A., Doglioli, A. M., Aldebert, C., Martinez, E., Koenig, G., Bonnet, S., and Moutin, T. (2020). The delayed island mass effect : How islands can remotely trigger blooms in the oligotrophic ocean. *Geophysical Research Letters*, 47(2) :e2019GL085282.
- [Millot and Taupier-Letage, 2005] Millot, C. and Taupier-Letage, I. (2005). Circulation in the Mediterranean Sea. In Salot, A., editor, *The Mediterranean Sea*, volume The Mediterranean Sea of *Handbook of Environmental Chemistry*, pages 29–66. Springer.
- [Qiu et al., 2011] Qiu, Z., Doglioli, A. M., He, Y., and Carlotti, F. (2011). Lagrangian model of zooplankton dispersion : numerical schemes comparisons and parameter sensitivity tests. *Chinese Journal of Oceanology and Limnology*, 29(2) :438–445.

Annexes

Annexe I : Script advection-diffusion

```
clear all ;
close all;
u=1 ; %vecteur u
v=0;  %vecteur v
n=25;%nombre particules
nd=86400;%nombre déplacements (temps en secondes)
x=zeros(n,nd);
y=zeros(n,nd);
alpha=(2*pi)/n;
x0=0;
y0=0;
for j=1:n %changement particules
    xdep=x0+500*cos(j*alpha);
    ydep=y0+500*sin(j*alpha);
    x(j,1)=xdep;
    y(j,1)=xdep;
    for i=1:nd %déplacements
        muxy=randn(1,2); %composante diffusive
        x(j,i+1)=x(j,i)+u+muxy(1);
        y(j,i+1)=y(j,i)+v+muxy(2);
    end
    plot(x(j,:),y(j,:))
    hold on
    plot(x(j,end),y(j,end),'*', 'markersize',10)
    hold on
end
```

Annexe II : Script pour un champ de vitesses stationnaire

```
clear global ;
close all;
obs="dataset-duacs-nrt-medsea-merged-allsat-phy-14_1587134940907.nc";
% obs possede des informations sur 10 jours
% donc nombre de jour en mer = 10 maximum
% -----Extraction des données-----
m_proj('utm','lon',[4.5 7.5],'lat',[42 43.8]);
lat=ncread(obs,'latitude');
lon=ncread(obs,'longitude');
vobs=ncread(obs,'vgos');
uobs=ncread(obs,'ugos');
time=ncread(obs,'time');
time=double(time);
time0=datetime(1950,01,01);
temps=time0+time./(1);
date=datestr(temps,24);
%-----Insertion des variables terrain-----%
resospa=0.125;
resotempo=3600*24;
londep0=6.16; %input('longitude en systeme WGS86 ');
latdep0=42.8; %input('latitude en systeme WGS86 ');
tempsj=5; %input('nombre de jour en mer (10 max) ');
t=fix(tempsj*(86400/resotempo));
%-----Carte courant Nord-----%
% [lonq,latq]=meshgrid(lon,lat);
% figure ()
% m_proj('utm','lon',[3 11],'lat',[42 44.6]);
% m_quiver(lonq,latq,uobs(:,1),vobs(:,1),0);
% m_gshhs_h('patch',[.5 .5 .5]);
% m_grid('box','fancy','tickdir','in');
% hold on
% m_quiver(4,43.75,0.5,0,0,'b')
% m_text(4.25,43.8,'0.5 m/s');
% m_text(5.98,43.2,'TOULON');
% m_text(5.44,43.34,'MARSEILLE');
% m_text(7.24,43.8,'NICE');
% m_text(9.44,45,'GENES');
% m_text(8.9,42.25,'CORSE');
% m_text(6.2,42.85,'Station \bf{ANTARES}');
% m_text(7.9,43.42,'Station \bf{DYFAMED}');
% m_line(6.16,42.8,'marker','square','markersize',8,'color','r');
% m_line(7.87,43.39,'marker','square','markersize',8,'color','r');
% m_line(5.92,43.12,'marker','square','markersize',6,'color','g');
% m_line(5.39,43.29,'marker','square','markersize',6,'color','g');
% m_line(7.2,43.71,'marker','square','markersize',6,'color','g');
% m_line(8.95,44.4,'marker','square','markersize',6,'color','g');
% m_northarrow(10.5,44,.5);
% hold on

%-----fond de carte-----#
figure()
[lonp,latp]=meshgrid(lon(14:37),lat(9:23));
m_quiver(lonp,latp,uobs(14:37,9:23,1),vobs(14:37,9:23,1),0);
m_gshhs_h('patch',[.5 .5 .5]);
m_grid('box','fancy','tickdir','in');
m_ruler([.5 .8],.1,3,'fontsize',10);
m_northarrow(6.5,43.5,.3);
hold on
%-----Positions de pts de repères-----%
m_quiver(5.3,43.5,0.1,0,0,'b')
m_text(5.3,43.6,'0.1 m/s')
m_text(5.98,43.15,'TOULON');
m_text(5.44,43.34,'MARSEILLE');
m_text(7.43,7,'NICE');
m_line(5.92,43.12,'marker','square','markersize',5,'color','r');
m_line(5.39,43.29,'marker','square','markersize',5,'color','r');
m_line(7.2,43.71,'marker','square','markersize',5,'color','r');
m_line(londep0,latdep0,'marker','square','markersize',4,'color','r');
m_text(londep0+0.13,latdep0,'\bf{ANTARES}');
hold on
%-----Calcul des trajectoires-----#
n=50;%nombre particules
x=zeros(n,t);
y=zeros(n,t);
alpha=(2*pi)/n;% angle pour le positionnement des points de départ
for ip=1:n %changement particules
    londep=londep0+0.125*cos(ip*alpha); % position x des pts
    latdep=latdep0+0.125*sin(ip*alpha); % position y des pts
    [x0,y0]=m_ll2xy(londep,latdep); %coord en m dans le ref utm
    x(ip,1)=x0; % valeur lon de départ en m
    y(ip,1)=y0; % valeur lat de départ en m
    life(ip)=1; %état de la particule(vivante si >0;morte si<0 ou =NaN)
    for it=1:t %déplacements en seconde
        %-----trouver la valeur de u et v à chaque t-----%
        [longit,latit]=m_xy2ll(x(ip,it),y(ip,it));%conversion xy->ll
        Plon= round((longit-lon(1))/(resospa)); %ligne de u et v
        Plat= round((latit-lat(1))/(resospa)); %colonne de u et v
        life(ip)=Plon*Plat; % test si la particule est hors limite
        if life(ip)>0 && ~isnan(life(ip)) %si ip reste dans les limites
            u=uobs(Plon,Plat,1); %valeur de u dans uobs en m/s
            v=vobs(Plon,Plat,1); %valeur de v dans vobs en m/s
            %-----Calcul des positions-----%
            muxy=randn(1,2); % composante diffusive
            x(ip,it+1)=x(ip,it)+resotempo*(0.01*muxy(1)+u);% coord suivant en m
            y(ip,it+1)=y(ip,it)+resotempo*(0.01*muxy(2)+v);% coord suivant en m
        else % condition si ip sort des limites
            it=t;
        end
    end
end
[nlon,nlat]=m_xy2ll(x,y); % conversion xy en ll des positions
for i=1:n
    m_plot(nlon(i,:),nlat(i,:),r'); % graph des trajectoire
    m_plot(nlon(i,end),nlat(i,end),r*','markersize',2);% position finale
    m_plot(nlon(i,1),nlat(i,1),r*','markersize',2);% position initiale
    hold on
end
```

Annexe III : Script pour un champ de vitesses variable

```
clear all ;
close all;
obs="dataset-duacs-nrt-medsea-merged-allsat-phy-14_1587134940907.nc";%2019
%obs="dataset-duacs-nrt-medsea-merged-allsat-phy-14_1587051233522.nc";%2018
%obs="dataset-duacs-nrt-medsea-merged-allsat-phy-14_1587051252803.nc";%2017
% obs possède des informations sur 10 jours
% donc nombre de jour en mer = 10 maximum
%-----Extraction des données----#
m_proj('utm','lon',[5 7.5],'lat',[42 43.8]);
lat=ncread(obs,'latitude');
lon=ncread(obs,'longitude');
v2019=ncread(obs,'vgos');
u2019=ncread(obs,'ugos');
time=ncread(obs,'time');
time=double(time);
time0=datetime(1950,01,01);
temps=time0+time./(1);
date=datestr(temps,24);
%----résolution spatio-temporelle----#
resotempo=100;
resospa=0.125;
londep0=6.16;%input('longitude en systeme WGS86 ');
latdep0=42.8;%input('latitude en systeme WGS86 ');
tempsj=5;%input('nombre de jour en mer (10 max) ');
t=fix(tempsj*(86400/resotempo));

%-----fond de carte----#
figure()
[lonp,latp]=meshgrid(lon(17:37),lat(9:23));
m_quiver(lonp,latp,u2019(17:37,9:23,1),v2019(17:37,9:23,1),0);
m_gshhs_h('patch',[.5 .5 .5]);
m_grid('box','fancy','tickdir','in');
m_ruler([.5 .8],.1,3,'fontsize',10);
m_northarrow(6.5,43.5,.3);
hold on
%-----Positions de pts de repères-----%
m_quiver(5.3,43.5,0.1,0,0,'b')
m_text(5.3,43.6,'0.1 m/s')
m_text(5.98,43.15,'TOULON');
m_text(5.44,43.34,'MARSEILLE');
m_text(7.43,7,'NICE');
m_line(5.92,43.12,'marker','square','markersize',5,'color','r');
m_line(5.39,43.29,'marker','square','markersize',5,'color','r');
m_line(7.2,43.71,'marker','square','markersize',5,'color','r');
%---point de départ---%
m_line(londep0,latdep0,'marker','square','markersize',4,'color','r');
m_text(londep0+0.13,latdep0,'\bf{ANTARES}');
hold on

%---Calcul des trajectoires-----#
n=50;%nombre particules
x=zeros(n,t);
y=zeros(n,t);
alpha=(2*pi)/n; % angle pour le positionnement des points de départ
for ip=1:n %changement particule
    k=1;%numéro du champ de vitesse
    londep=londep0+0.125*cos(ip*alpha); % position x des pts de la tache
    latdep=latdep0+0.125*sin(ip*alpha); % position y des pts de la tache
    [x0,y0]=m_ll2xy(londep,latdep); %coord en m dans le ref utm
    x(ip,1)=x0; % valeur lon de départ en m
    y(ip,1)=y0; % valeur lat de départ en m
    life(ip)=1; %état de la particule(vivante si >0; morte si<0 ou=NaN)
    for it=1:t %déplacements en seconde
        %---trouver la valeur de u et v à chaque t---%
        [longit,latit]=m_xy2ll(x(ip,it),y(ip,it)); % Conversion xy->ll
        Plon= round((longit-lon(1))/(resospa)); %ligne de u et v
        Plat= round((latit-lat(1))/(resospa)); %colonne de u et v
        life(ip)=Plon*Plat; % test si la particule est hors limite ou non
        if (life(ip)>0 && ~isnan(life(ip))) % si ip est dans les limites
            u=u2019(Plon,Plat,k); %valeur de u dans u2019 en m/s
            v=v2019(Plon,Plat,k); %valeur de v dans v2019 en m/s
            %----Calcul des positions----%
            muxy=randn(1,2); % composante diffusive
            x(ip,it+1)=x(ip,it)+resotempo*(0.01*muxy(1)+u);% coord suivant en m
            y(ip,it+1)=y(ip,it)+resotempo*(0.01*muxy(2)+v);% coord suivant en m
        else % condition si ip sort des limites
            it=t;
        end
        if it==k*(86400/resotempo) % Changement de champ de vitesse
            k=k+1;
        else
            k=k;% garde le champ de vitesse
        end
    end
end
[nlon,nlat]=m_xy2ll(x,y); % conversion xy en ll des positions
for i=1:n
    m_plot(nlon(i,:),nlat(i,:),'r'); % graph des trajectoire
    m_plot(nlon(i,end),nlat(i,end),'r*','markersize',2);%position finale
    m_plot(nlon(i,1),nlat(i,1),'r*','markersize',2);%position de départ
    hold on
end
```

Annexe IV : Script pour la comparaison AVISO vs. MERCATOR

```
clear all ;
close all;
obs="dataset-duacs-nrt-medsea-merged-all1sat-phy-14_1587134940907.nc";%obs2019
modele="med00-cmcc-cur-an-fc-d_1587052977053.nc";%HR2019
% # obs possede des informations sur 10 jours
% # donc nombre de jour en mer = 10 maximum
% #-----Extraction des données-----#
% #-----Données observation-----#
m_proj('utm','lon',[4 7.5],'lat',[42.4 43.8]);
lat=ncread(obs,'latitude');
lon=ncread(obs,'longitude');
v2017=ncread(obs,'vgos');
u2017=ncread(obs,'ugos');
time=ncread(obs,'time');
time=double(time);
time0=datetime(1950,01,01);
temps=time0+time./(1);
date=datestr(temps,24);
%-----données modele-----#
latm=ncread(modele,'lat');
lonm=ncread(modele,'lon');
um2017=ncread(modele,'uo');
vm2017=ncread(modele,'vo');
timem=ncread(modele,'time');
time0=datetime(1950,01,01);
tempsm=timem0+timem./(1);
date=datestr(temps,24);
%-----résolution spatio-tempo et données terrain-----#
resotempo=60;
resospa=0.125;
resospam=0.042;
londep0=input("Longitude en systeme WGS86 ");
latdep0=input("Latitude en systeme WGS86 ");
tempsj=input("nombre de jour en mer (10 max) ");%temps de déplacement en jour
t=fix(tempsj*(86400/resotempo));
[lonq,latq]=meshgrid(lon,lat);
%-----carte de base-----#
figure(1)
[lonp,latp]=meshgrid(lon(9:36),lat(12:22));
m_quiver(lonp,latp,u2017(9:36,12:22,1)',v2017(9:36,12:22,1)',0);
m_gshhs_h('patch',[.5 .5 .5]);
m_grid('box','fancy','tickdir','in');
m_ruler([.5 .8],1,3,'fontsize',10);
m_northarrow(6.5,43.5,.3);
hold on

%-----points de repère-----#
m_quiver(5.3,43.5,0.1,0,0,'b')
m_text(5.3,43.6,'0.1 m/s');
m_text(5.98,43.15,'TOULON');
m_text(5.44,43.34,'MARSEILLE');
m_text(7.43,7,'NICE');
m_line(5.92,43.12,'marker','square','markersize',5,'color','r');
m_line(5.39,43.29,'marker','square','markersize',5,'color','r');
m_line(7.2,43.71,'marker','square','markersize',5,'color','r');
m_line(londep0,latdep0,'marker','square','markersize',4,'color','r');
m_text(londep0+0.13,latdep0,'\bf{ANTARES}');
hold on

%-----Calcul des trajectoires-----#
n=50;%nombre particule
x=zeros(n,t);
y=zeros(n,t);
alpha=(2*pi)/n;
%-----Boucle « AVISO »-----#
for ip=1:n %changement particule
    k=1;
    londep=londep0+0.125*cos(ip*alpha); % position x des pts
    latdep=latdep0+0.125*sin(ip*alpha); % position y des pts
    [x0,y0]=m_ll2xy(londep,latdep); %coord en m dans le ref utm
    x(ip,1)=x0; % valeur lon de départ en m
    y(ip,1)=y0; % valeur lat de départ en m
    life(ip)=1;
    for it=1:t %déplacements en seconde
        [longit,latit]=m_xy2ll(x(ip,it),y(ip,it)); % conversion xy->ll
        Plon= fix((longit-lon(1))/(resospa)); %ligne de u et v
        Plat= fix((latit-lat(1))/(resospa)); %colonne de u et v
        life(ip)=Plon*Plat; % test si la particule est hors limite
        if (life(ip)>0 && ~isnan(life(ip))) % condition si ip reste dans les limites
            u=u2017(Plon,Plat,k); %valeur de u dans u2017 en m/s
            v=v2017(Plon,Plat,k); %valeur de v dans v2017 en m/s
            %-----Calcul des positions-----#
            muxy=randn(1,2); % composante advective
            x(ip,it+1)=x(ip,it)+resotempo*(0.01*muxy(1)+u); % coord suivant en m
            y(ip,it+1)=y(ip,it)+resotempo*(0.01*muxy(2)+v); % coord suivant en m
        else % condition si ip sort des limites
            it=t;
        end
        if it==k*(86400/resotempo) % changement de champ de vitesse
            k=k+1;
        else
            k=k;
        end
    end
end
[nlon,nlat]=m_xy2ll(x,y); % conversion xy en ll des positions
for i=1:n
    m_plot(nlon(i,:),nlat(i,:), 'r'); % graph des trajectoire
    m_plot(nlon(i,end),nlat(i,end), 'r','markersize',2); % position finale
    m_plot(nlon(i,1),nlat(i,1), 'r','markersize',2); % position initiale
    hold on
end
```



```

%-----boucle « MERCATOR »-----#

x=zeros(n,t);
y=zeros(n,t);
for ip=1:n %changement particules
    k=1;
    londep=londep0+0.125*cos(ip*alpha); % position x des pts de la tache
    latdep=latdep0+0.125*sin(ip*alpha); % position y des pts de la tache
    [x0,y0]=m_ll2xy(londep,latdep); %coord en m dans le ref utm
    x(ip,1)=x0; % valeur lon de départ en m
    y(ip,1)=y0; % valeur lat de départ en m
    life(ip,1)=1;
    for it=1:t %déplacements en seconde
        [longit,latit]=m_xy2ll(x(ip,it),y(ip,it)); % Conversion xy->ll
        Plon= round((longit-lonm(1))/(resospam)); %ligne de u et v
        Plat= round((latit-latm(1))/(resospam)); %colonne de u et v
        life(ip,it)=Plon*Plat; % test si la particule est hors limite
        if (life(ip,it)>0 && ~isnan(life(ip,it)))%si ip reste dans les limites
            u=um2017(Plon,Plat,1,k); %valeur de u dans u2019 en m/s
            v=vm2017(Plon,Plat,1,k); %valeur de v dans v2019 en m/s
            %-----Calcul des positions-----%
            muxy=randn(1,2); % composante advective
            x(ip,it+1)=x(ip,it)+resotempo*(0.01*muxy(1)+u);%coord suivant(m)
            y(ip,it+1)=y(ip,it)+resotempo*(0.01*muxy(2)+v);%coord suivant(m)
        else % condition si ip sort des limites
            it=t;
        end
        if it==k*(86400/resotempo) % Changement de champ de vitesse
            k=k+1;
        else
            k=k;
        end
    end
end
[nlon,nlat]=m_xy2ll(x,y); % conversion xy en ll des positions
for i=1:n
    m_plot(nlon(i,:),nlat(i,:), 'k'); % graph des trajectoires
    m_plot(nlon(i,end),nlat(i,end), 'k*', 'markersize',2);%position finale
    m_plot(nlon(i,1),nlat(i,1), 'k*', 'markersize',2);%position initiale
    hold on
end

```

Résumé

Les chercheurs du MIO ont prévu une campagne de prélèvement de neige marine vers la station océanographique ANTARES. La méthode est de laisser dériver des pièges à sédiments sur des lignes dérivantes. La station ANTARES se trouve sur le passage du courant Nord. De ce fait, cela nous suggère de développer un algorithme afin de prévoir la trajectoire des bouées sur 4 à 5 jours. Ceci à l'aide de champs de vitesses issus de données satellites (AVISO) et des données de modèles (MERCATOR). Les trajectoires sont simulées grâce à un modèle d'advection-diffusion à particules Lagrangiennes. Celles-ci sont simulées dans un repère géographique en deux dimensions, longitudes et latitudes. Les trajectoires obtenues peuvent être considérées comme le scénario le plus pessimiste. Les incertitudes sont trop importantes pour en déduire avec fiabilité une aire de récupération des bouées déployées. Il apparaît ici que pour affiner les simulations nous avons besoin de considérer une dérive en trois dimensions tout en intégrant la forme de la ligne dérivante.

Mots-clés : Station ANTARES, Courant Nord, Données AVISO, Données MERCATOR, Dérive Lagrangienne, Advection-diffusion, Simulation de trajectoires.

Abstract

MIO researchers are planning to sample marine snow around the ANTARES oceanographic station. The methodology is to use sediment traps on drifting buoys. Due to the ANTARES station's location in the North current, we are required to develop an algorithm to predict the paths of the drifting buoys over a 4 to 5 day period. This was done using current fields from satellite data (AVISO) and modeling data (MERCATOR). The path simulations are obtained by an advection-diffusion Lagrangian particles model. This are simulate in two dimensional geographic coordinate system, longitude, latitude. We can consider the path simulate as the most pessimistic scenario. The models are not reliable enough to predict an area to recover the buoys. To refine the simulation, we need to consider a path in three dimensions and integrate the shape of the buoys.

Keywords : ANTARES station, North current, AVISO data, MERCATOR data, Lagrangian particles, Advection-diffusion, Paths simulation.