Sommaire

[1. Récupération des données 3](#_Toc471983912)

[2. Librairies MATLAB à installer 3](#_Toc471983913)

[3. Saisie des paramètres spécifiques au mouillage et aux ADCP 4](#_Toc471983914)

[3.1. Saisie des paramètres du mouillage 4](#_Toc471983915)

[3.2. Saisie des paramètres de l’ADCP 4](#_Toc471983916)

[3.3. Déviation magnétique moyenne 5](#_Toc471983917)

[3.4. Lecture du fichier 5](#_Toc471983918)

[3.5. Correction de la profondeur de l’ADCP 6](#_Toc471983919)

[3.6. Indices pour lesquels l’ADCP est stabilisé 7](#_Toc471983920)

[3.7. Exclusion des données de mauvaise qualité (Pg field) 7](#_Toc471983921)

[4. Traitement des données ADCP 8](#_Toc471983922)

[4.1. Correction de la déviation magnétique 8](#_Toc471983923)

[4.2. Suppression des données de mauvaise qualité (Pg field) 8](#_Toc471983924)

[4.3. Calcul et correction de la profondeur de chaque bin 8](#_Toc471983925)

[4.4. Suppression des données de mauvaise qualité 10](#_Toc471983926)

[5. Interpolation et ré-échantillonnage des données 12](#_Toc471983927)

[5.1. Interpolation sur une grille verticale régulière 12](#_Toc471983928)

[5.2. Interpolation sur une grille horizontale régulière 12](#_Toc471983929)

[5.3. Filtre de la marée 12](#_Toc471983930)

[5.4. Ré-échantillonnage 12](#_Toc471983931)

[6. Combinaison des ADCP ‘up’ et ‘down’ 14](#_Toc471983932)

[7. Calcul du backscatter relatif 16](#_Toc471983933)

[8. Jeu de données global 16](#_Toc471983934)

[8.1. Ajouter un jeu de données à la série temporelle 16](#_Toc471983935)

[8.2. Création du fichier NetCDF global 17](#_Toc471983936)

[9. Bibliographie 18](#_Toc471983937)

[10. Suivi des versions de ce document 19](#_Toc471983938)

Aucun logiciel spécifique n’est requis

# Récupération des données

Cf. PROTOCOLE\_UTILISATION\_RDI\_ADCP.docx

Récupérer le fichier .000 après récupération de l’ADCP.

Les fichiers .000 contiennent les données brutes enregistrées par l’ADCP.

# Librairies MATLAB à installer

Les fonctions MATLAB suivantes sont nécessaires au traitement des données ADCP :

* adcp\_check\_surface.m
* adcp\_shadowzone.m
* adcp\_surface\_fit.m
* gregorian.m
* gregtick.m
* hamm.m
* hammfilter\_nodec.m
* julian.m
* meanmiss.m
* nmedian.m
* read\_os3.m
* sw\_dpth.m
* uvrot.m

Copier les répertoires ‘moored\_adcp\_proc’ et ‘volume\_backscatter’ situés à <ftp://ftp.ifremer.fr/ifremer/ird/jhabasqu/PIRATA_adcp_mooring/program/> dans votre espace de travail MATLAB.

# Saisie des paramètres spécifiques au mouillage et aux ADCP

Ouvrir le programme Matlab ‘template\_get\_adcp\_data.m’

Définir le chemin vers les librairies MATLAB utilisées :

|  |
| --- |
| % path  addpath('C:\Workspace\_Matlab\ADCP\_mooring\moored\_adcp\_proc');  addpath('C:\Workspace\_Matlab\ADCP\_mooring\volume\_backscatter'); |

## Saisie des paramètres du mouillage

* Nom
* Latitude
* Longitude

|  |
| --- |
| mooring.name='KPO\_1140';  mooring.lat=0;  mooring.lon=-23; |

## Saisie des paramètres de l’ADCP

Définir l’emplacement du fichier :

|  |
| --- |
| % Location rawfile  rawfile='QM14911/DPL1\_000.000'; |

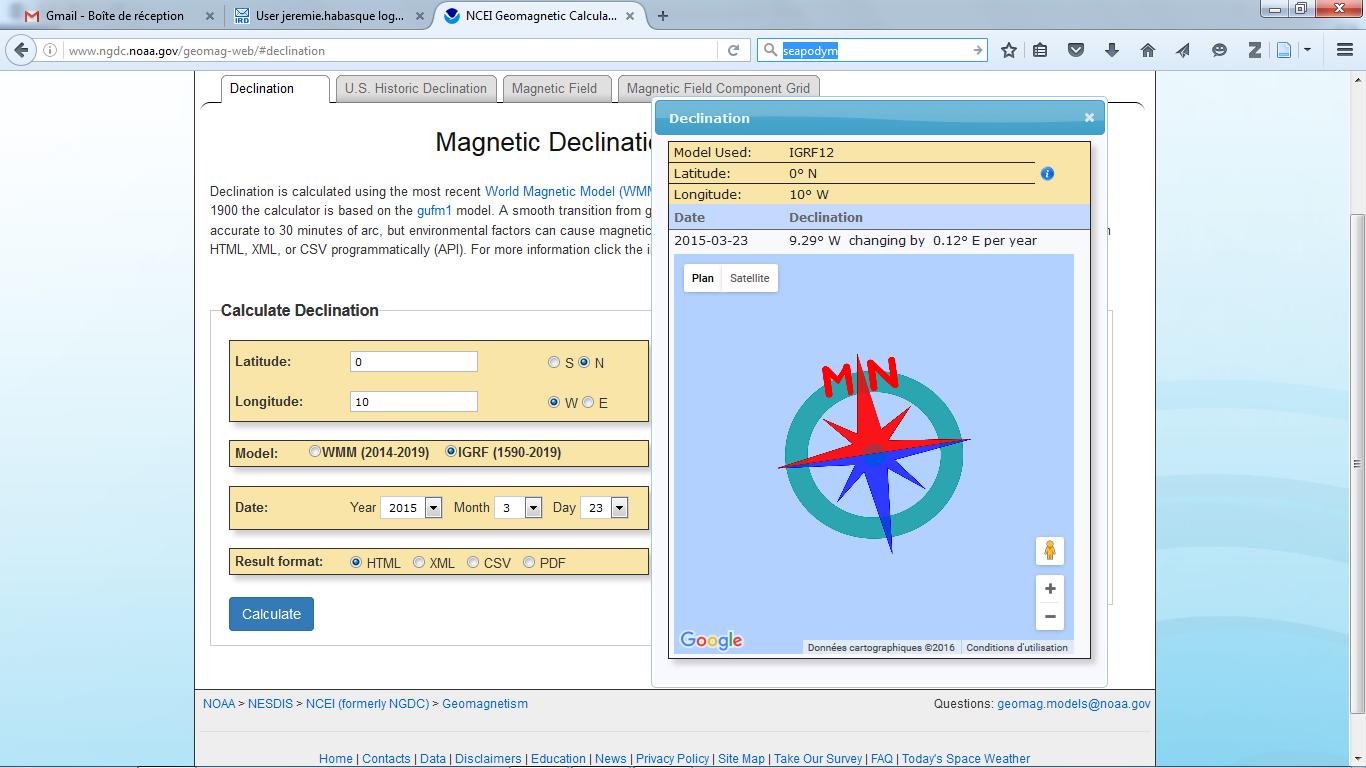
* Numéro de série
* Type : ‘Quartermaster’, ‘longranger’
* Direction : ‘up’ ou ‘dn’
* Profondeur nominale de l’instrument
* Numéro de l’instrument

|  |
| --- |
| adcp.sn= 15258;  adcp.type= ‘150 khz Quartermaster’;  adcp.direction='up';  adcp.instr\_depth=300;  instr = 1; |

## Déviation magnétique moyenne

Si l’ADCP n’a pas été configuré pour corriger la déviation magnétique (code "EA0" dans le fichier de configuration »), utilisez <http://www.ngdc.noaa.gov/geomag-web/#declination> pour calculer les déviations aux dates de déploiements et de récupération du mouillage.

Sélectionner le modèle IGRF.



Saisir les valeurs obtenues pour calculer la déviation magnétique moyenne (en degrés) :

|  |
| --- |
| % magnetic deviation values  magnetic\_deviation\_ini = 15.11;  magnetic\_deviation\_end = 15.01; |

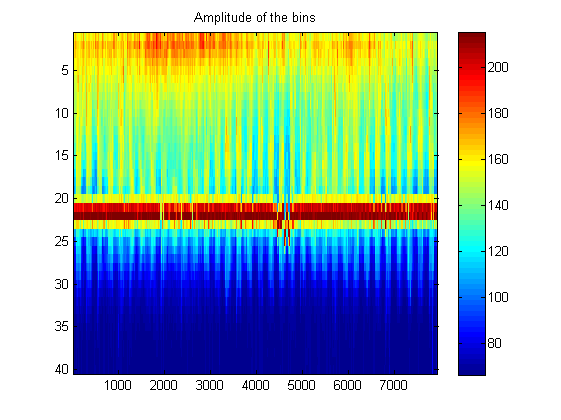
## Lecture du fichier

## Correction de la profondeur de l’ADCP

Les données du capteur de pression sont régulièrement fausses. Si l’ADCP est orienté vers la surface, une correction de la profondeur de l’instrument peut alors être effectuée en fonction de la surface reflection.

L’intervalle des bins qui couvrent la surface reflection de l’ADCP peut être trouvé en traçant l’amplitude des échos.

|  |
| --- |
| ea = squeeze(mean(raw.amp(:,:,first:last),2)); % amplitude of the bins  figure; imagesc(ea);title('Amplitude of the bins'); colorbar |



Les fortes amplitudes sont localisées où est la surface reflection. Choisissez un intervalle de sorte que la surface reflection est couverte sur l’ensemble de la série (même lorsque le mouillage plonge). A chaque pas de temps, une sélection plus fine de cet intervalle est ensuite effectuée par le programme.

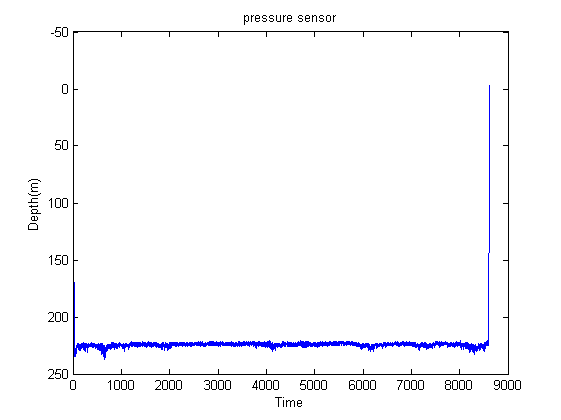
Sur l’exemple ci-dessus, autour de l’index 4700 il y a un certain nombre de profiles où l’amplitude maximum des échos se situent au bin 25 ou plus. Ici l’intervalle 18 à 28 est un bon choix.

|  |
| --- |
| sbins=[18:28]; |

## Indices pour lesquels l’ADCP est stabilisé

Les indices de l’instrument à profondeur d’acquisition sont déterminés en traçant la série temporelle de la pression :

|  |
| --- |
| plot(raw.pressure);set(gca,'ydir','reverse');title('pressure sensor');ylabel('Depth(m)');xlabel('Time'); |



Saisir les indices de début et fin :

|  |
| --- |
| first = 21;  last = 8605; |

## Exclusion des données de mauvaise qualité (Pg field)

Pour supprimer les données dont le pourcentage de données est bon, il faut utiliser le champ ‘Pg’ du 4ème transducteur.

|  |
| --- |
| % Exclude data with percent good below prct\_good  prct\_good = 20; |

# Traitement des données ADCP

## Correction de la déviation magnétique

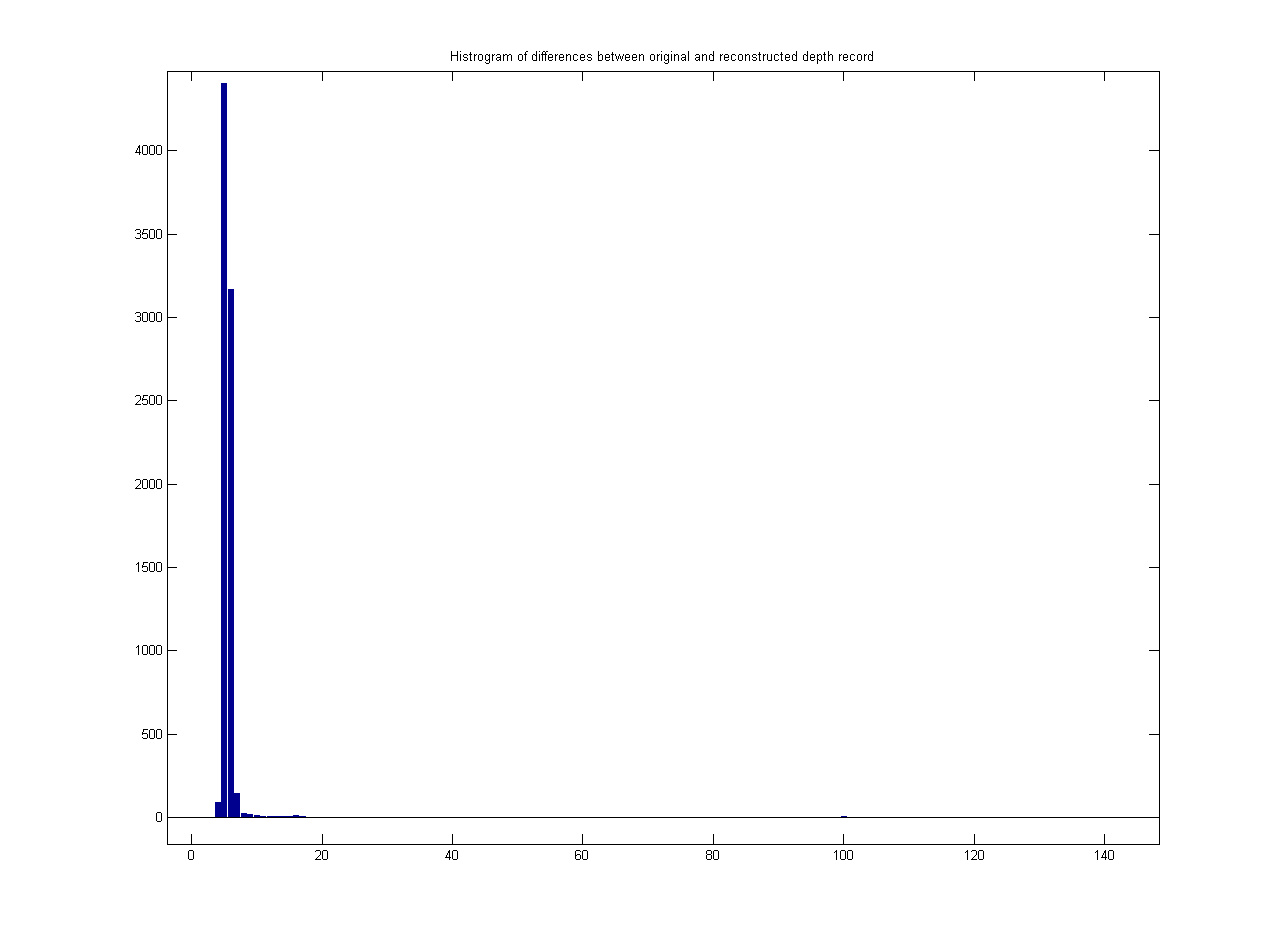
La correction de la déviation magnétique est effectuée par la fonction ‘uvrot.m’

## Suppression des données de mauvaise qualité (Pg field)

## Calcul et correction de la profondeur de chaque bin

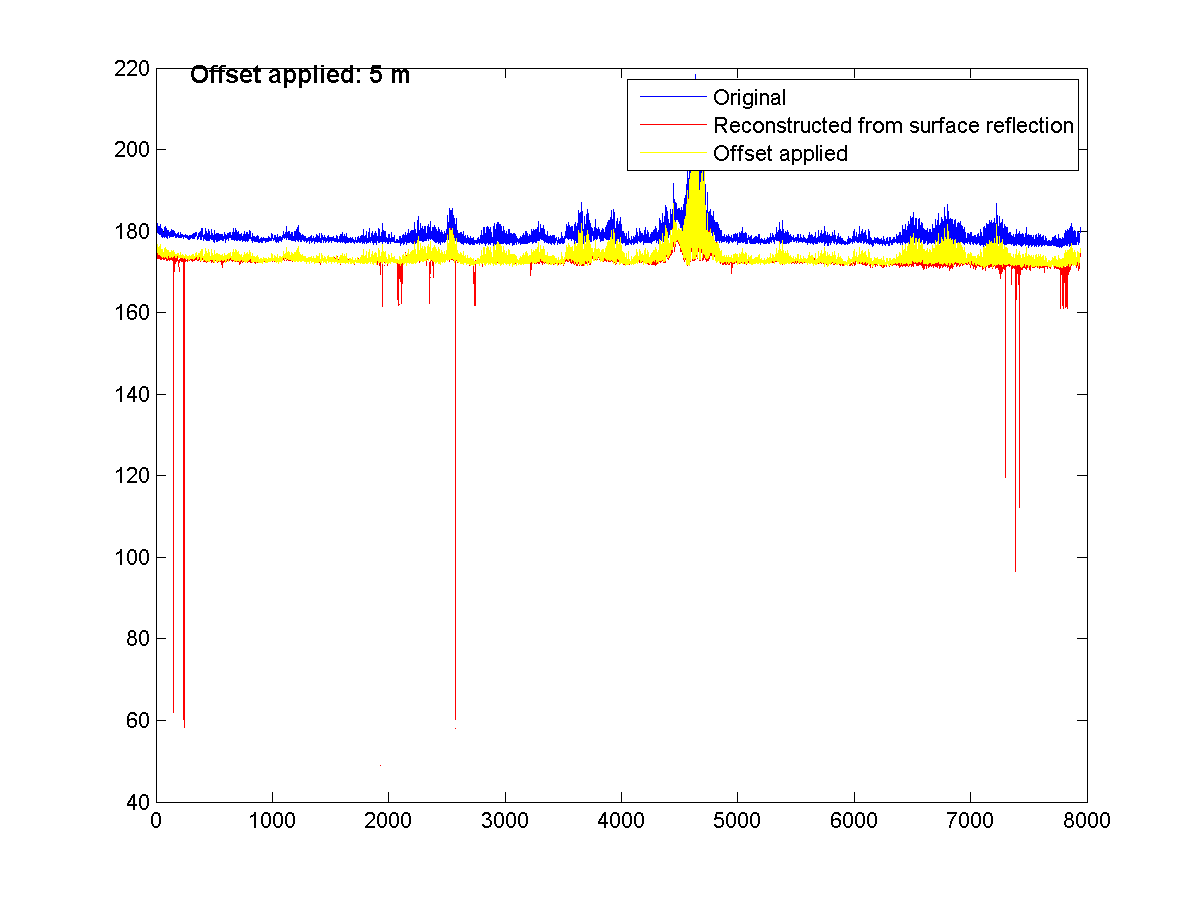
La profondeur de chaque bin est calculée à partir de la pression avec la fonction ‘sw\_dpth’.

La correction de la profondeur de chaque bin en fonction de l’écho de surface est réalisée par la fonction ‘adcp\_surface\_fit. L’offset à appliquer à chaque profondeur de bin est calculé. Il est également possible de saisir un offset manuellement.



L’offset appliqué est vérifié en traçant l’histogramme des différences entre l’enregistrement original et la profondeur reconstruite. Une distribution unimodale de l’offset est nécessaire.

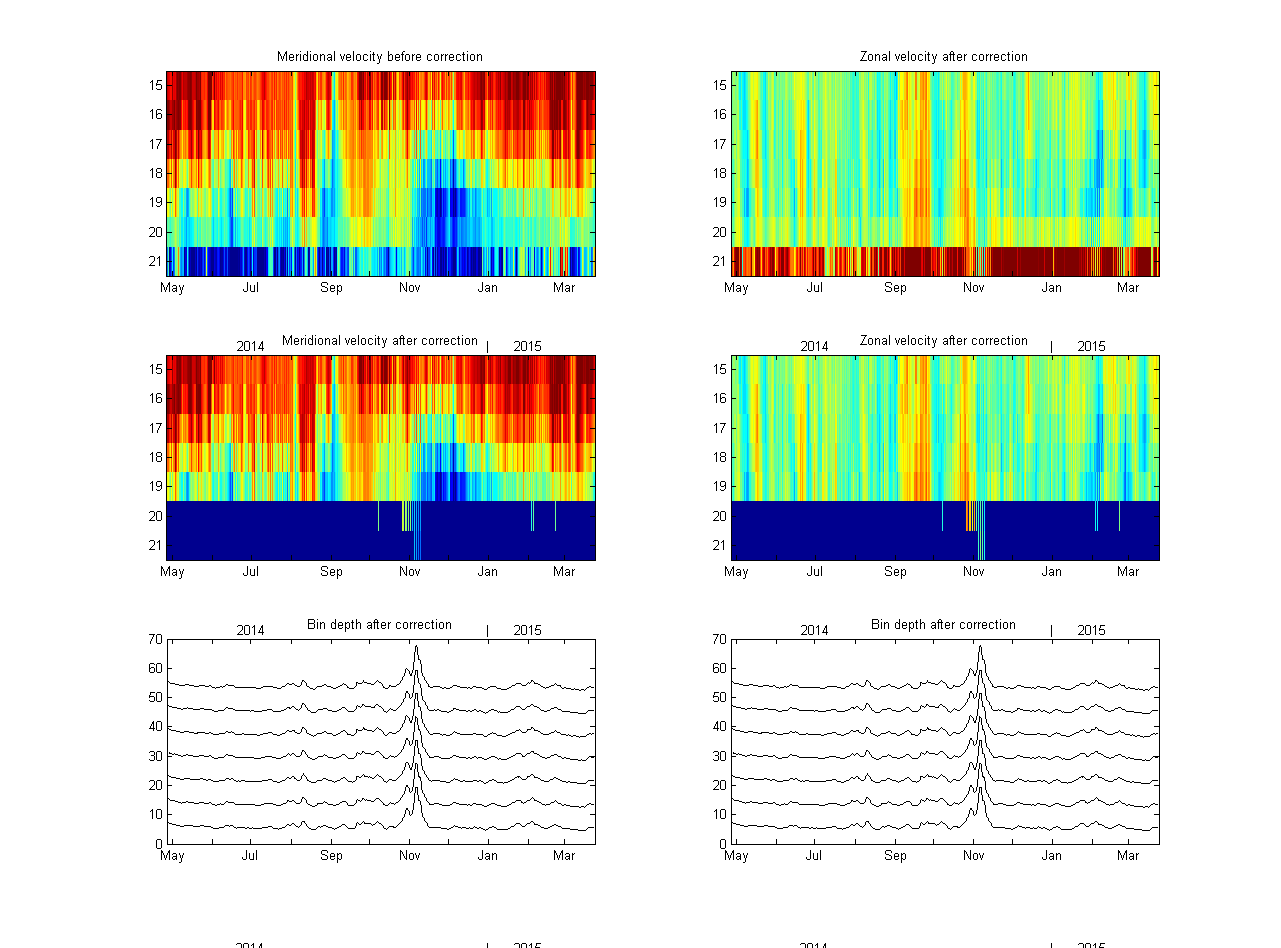
L’offset est alors appliqué pour obtenir des profondeurs correctes de l’instrument et des bins.



## Suppression des données de mauvaise qualité

Si l’ADCP est orienté vers la surface, la profondeur sous la surface invalide due à la surface reflection est calculée (*adcp\_shadowzone.m*). Cette ‘shadow zone’ est déterminée par la profondeur de l’instrument et l’angle du faisceau.

Les bins proches de la surface supposés avoir des vitesses de courant correctes sont représentés (*adcp\_check\_surface.m*). S’il y a toujours des mauvaises vitesses de courant, un critère manuel doit être trouvé.



Les données sont enregistrées dans une matrice sous la forme ‘mooring\_name, serial number, instrument number.mat’. (exemple : 23W-0N\_14911\_instr\_01.mat)

# Interpolation et ré-échantillonnage des données

Etapes d’interpolation

## Interpolation sur une grille verticale régulière

Puisque les profondeurs du centre des « bins » varient pour chaque profil, les données de vitesse sont interpolées à des profondeurs standards fonction de la longueur de bin.

## Interpolation sur une grille horizontale régulière

Les données sont interpolées sur une grille temporelle régulière.

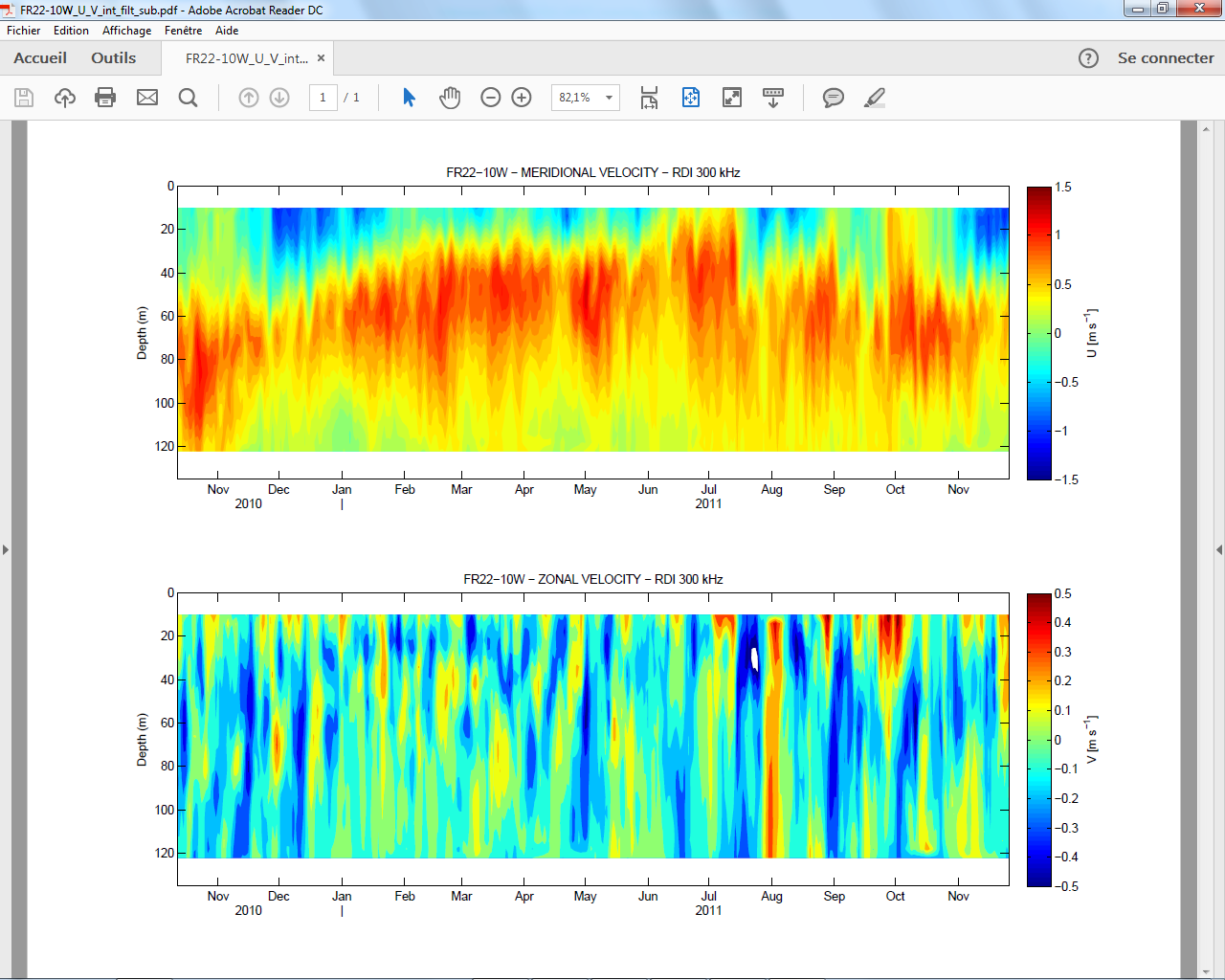
## Filtre de la marée

Sur les mouillages ADCP, l’effet barotropique est faible comparé à l’effet baroclinique. Pour supprimer l’effet de la marée de la série de données, un filtre basse fréquence (*mfilter.m*).

Comme intervalle de sous-échantillonnage pour le filtre basse fréquence, nous avons choisi 40 heures.

## Ré-échantillonnage

Les jeux de données sont alors ré-échantillonnés sur une grille régulière d’un pas de 12 heures centrée sur midi (minuit-midi-minuit, etc.)



La figure des données interpolées est sauvegardée sous le nom ‘mooring\_name\_ \_U\_V\_int\_filt\_sub.pdf’

Les données interpolées sont enregistrées dans une matrice sous la forme « mooring\_name, serial number, instrument number\_int\_filt\_sub.mat ».

Exemple : 23W-0N\_14911\_instr\_01\_int\_filt\_sub.mat

# Combinaison des ADCP ‘up’ et ‘down’

Dans le programme ‘template\_get\_adcp\_data\_combine.m’, charger les matrices interpolées, filtrées et ré-échantillonnées des ADCP up et down :

|  |
| --- |
| load([fpath, '23W-0N\_14911\_instr\_01\_int\_filt\_sub.mat']);  adcp\_up=adcp;  up\_u=data.uintfilt;  up\_v=data.vintfilt;  up\_z=data.Z;  up\_time=data.inttim;  npts\_up= data.npts\_up;  freq\_up=adcp\_up.config.sysconfig.frequency;    load([fpath, '23W-0N\_2627\_instr\_02\_int\_filt\_sub.mat']);  adcp\_down=adcp;  down\_u=data.uintfilt;  down\_v=data.vintfilt;  down\_z=data.Z;  down\_time=data.inttim;  freq\_down = adcp\_down.config.sysconfig.frequency; |

Les ADCP orientés vers le haut et le bas sont installés à 3 mètres d’écart. Il faut donc calculer la distance entre la première mesure de l’ADCP up et la première mesure de l’ADCP down.

Cette distance correspond à:

La moitié de la distance du premier bin de l’ADCP ‘up’

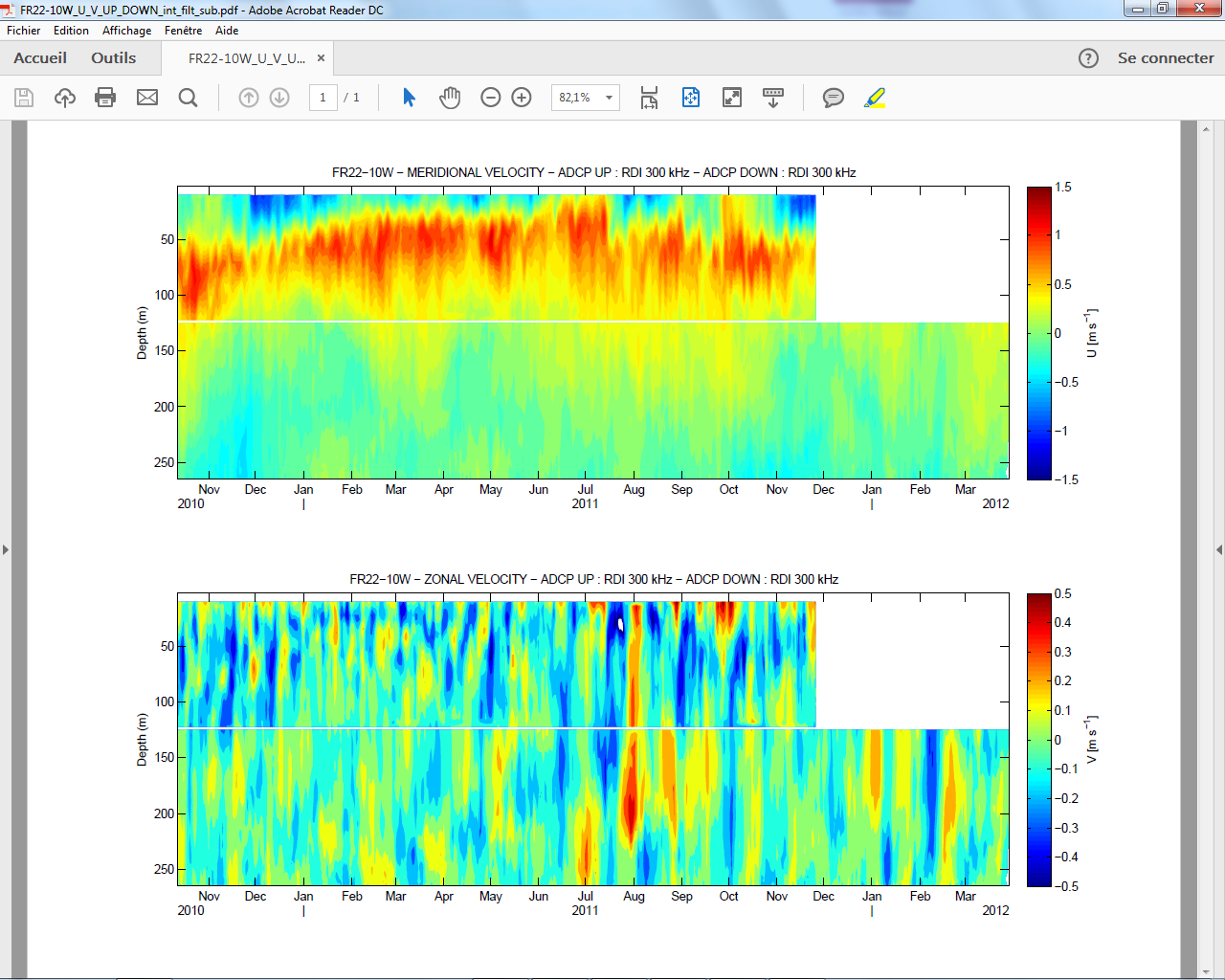
+ le ‘blank’ de l’ADCP ‘up’

+ la distance entre les 2 instruments

+ le ‘blank’ de l’ADCP ‘down’

+ la moitié de la distance du premier bin de l’ADCP ‘down’

|  |
| --- |
| distance\_between\_up\_and\_down = adcp\_up.config.cell/2 + adcp\_up.config.blank + distance\_between\_instruments + adcp\_down.config.blank + adcp\_down.config.cell/2; |



La figure finale est sauvegardée sous le nom ‘mooring\_name\_U\_V\_UP\_DOWN\_int\_filt\_sub.pdf’

Les données finales sont sauvegardées dans la matrice ‘mooring\_name \_U\_V\_UP\_DOWN\_int\_filt\_sub.mat’

# Calcul du backscatter relatif

Le calcul de ‘target strength’ à partir de l’amplitude d’échos de l’ADCP est effectué selon (Plimpton et al., 2004).

Les profils de salinité et de température (CTD) avant déploiement (ou récupération) sont utilisés comme paramètres de la fonction de calcul de ‘target strength’ (*targetstrength.m*).

# Jeu de données global

La grille régulière est définie de la surface jusqu’à 350 mètres par pas de 5 mètres.

## Ajouter un jeu de données à la série temporelle

* Ouvrir le script ‘merge\_PIRATA\_ADCP\_mooring\_name.m’
* Ajouter la lecture du nouveau jeu de données

|  |
| --- |
| % 2015-2016  load('C:\Users\jhabasqu\Desktop\PIRATA\ADCP\_mooring\_data\23W-0N\2015-2016 - kpo\_1140\23W-0N\_2627\_UP\_DOWN\_int\_filt\_sub.mat');  adcp\_2015\_time = data.inttim;  adcp\_2015\_u = data.u\_final;  adcp\_2015\_v = data.v\_final;  adcp\_2015\_z = data.z\_final; |

* Ajouter l’interpolation du nouveau jeu de données sur la grille verticale régulière

|  |
| --- |
| %interpolation for each timestep for 2015 data  u\_interp\_2015 = NaN(length(Z),length(adcp\_2015\_time));  v\_interp\_2015 = NaN(length(Z),length(adcp\_2015\_time));  for i=1:length(adcp\_2015\_time)  ind\_ok = find(~isnan(adcp\_2015\_u(:,i)));  u\_interp\_2015(:,i) = interp1(adcp\_2015\_z(ind\_ok),adcp\_2015\_u(ind\_ok,i),Z);  v\_interp\_2015(:,i) = interp1(adcp\_2015\_z(ind\_ok),adcp\_2015\_v(ind\_ok,i),Z);  end |

* Ajout des champs u, v interpolés et time au jeu de données global

|  |
| --- |
| %% combine all data  all\_time = [adcp\_2002\_time adcp\_2006\_time adcp\_2008\_time adcp\_2010\_time adcp\_2011\_time adcp\_2012\_time adcp\_2014\_time adcp\_2015\_time];  all\_u\_interp = [u\_interp\_2002 u\_interp\_2006 u\_interp\_2008 u\_interp\_2010 u\_interp\_2011 u\_interp\_2012 u\_interp\_2014 u\_interp\_2015];  all\_v\_interp = [v\_interp\_2002 v\_interp\_2006 v\_interp\_2008 v\_interp\_2010 v\_interp\_2011 v\_interp\_2012 v\_interp\_2014 v\_interp\_2015]; |

La figure finale est sauvegardée sous le nom ‘ADCP\_U\_V\_mooring\_name\_daily.pdf’.

Les données finales sont enregistrées dans une matrice sous la forme ‘mooring\_name, annee\_debut\_anne\_fin\_int\_filt\_sub.mat'

(Exemple : ADCP\_10W0N\_2001\_2015\_int\_filt\_sub.mat').

## Création du fichier NetCDF global

|  |
| --- |
| %% Write netcdf file  ncid=netcdf.create([fpath,'ADCP\_23W0N\_2001\_2016\_1d.nc'],'NC\_WRITE');    %create dimension  dimidt = netcdf.defDim(ncid,'time',length(ADCP\_final.time));  dimidz = netcdf.defDim(ncid,'depth',length(ADCP\_final.depth));  %Define IDs for the dimension variables (pressure,time,latitude,...)  time\_ID=netcdf.defVar(ncid,'time','double',[dimidt]);  depth\_ID=netcdf.defVar(ncid,'depth','double',[dimidz]);  %Define the main variable ()  u\_ID = netcdf.defVar(ncid,'u','double',[dimidt dimidz]);  v\_ID = netcdf.defVar(ncid,'v','double',[dimidt dimidz]);  %We are done defining the NetCdf  netcdf.endDef(ncid);  %Then store the dimension variables in  netcdf.putVar(ncid,time\_ID,ADCP\_final.time);  netcdf.putVar(ncid,depth\_ID,ADCP\_final.depth);  %Then store my main variable  netcdf.putVar(ncid,u\_ID,ADCP\_final.u);  netcdf.putVar(ncid,v\_ID,ADCP\_final.v);  %We're done, close the netcdf  netcdf.close(ncid); |

# Bibliographie

Plimpton, P.E., Freitag, H.P., and McPhaden, M.J. (2004). Processing of subsurface ADCP data in the Equatorial Pacific.

# Suivi des versions de ce document

|  |  |  |  |
| --- | --- | --- | --- |
| **Rédacteur** | | **Approbateur** | |
| Nom : | Habasque Jérémie | Nom : | J. Grelet |
| Fonction : | IE | Fonction : | Responsable Laboratoie Physique IRD Bretagne |

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Commentaires et modifications** |
| 06/12/2016 | 01 |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
| **Relecteur** | **Date** |
|  |  |