

# Traitement des données MEG avec Fieldtrip

Illustration avec le projet « CATSEM »  
(Catégorisation Sémantique)

C. Zielinski



**Jeudi 7 mai 2014**

10h30 Salle de conférences B011, bât. B LPL 5 avenue Pasteur, Aix-en-Provence

(Labex BLRI)

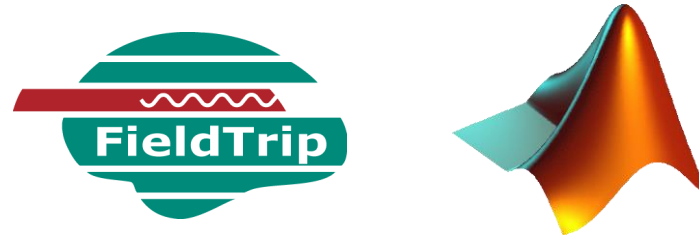
**Valérie Chanoine, Christelle Zielinski**

**Projet MEG Catsem - de la conception du design expérimental aux analyses statistiques pour le moyennage et la localisation des sources.**

*Sujet :*

Dans le contexte d'un projet BLRI réalisé en Magnéto-Encéphalographie (MEG), nous vous proposons une revue pratique des étapes courantes du traitement des données MEG. Après une brève description du design expérimental (tâche de décision sémantique sur présentation visuelle ou auditive de mots), nous nous appuierons sur deux logiciels couramment utilisés en MEG (à savoir « Fieldtrip » et « Brainstorm »), pour aborder de manière très pratique les thèmes du pré-traitement des données MEG (filtrage, nettoyage du signal), du moyennage (permettant l'obtention de champs évoqués), de la visualisation de l'activité cérébrale par topographie ou par la localisation de source. A cette occasion, nous vous présenterons également un outil statistique mis à disposition par le logiciel « Brainstorm » pour effectuer des comparaisons multiples.

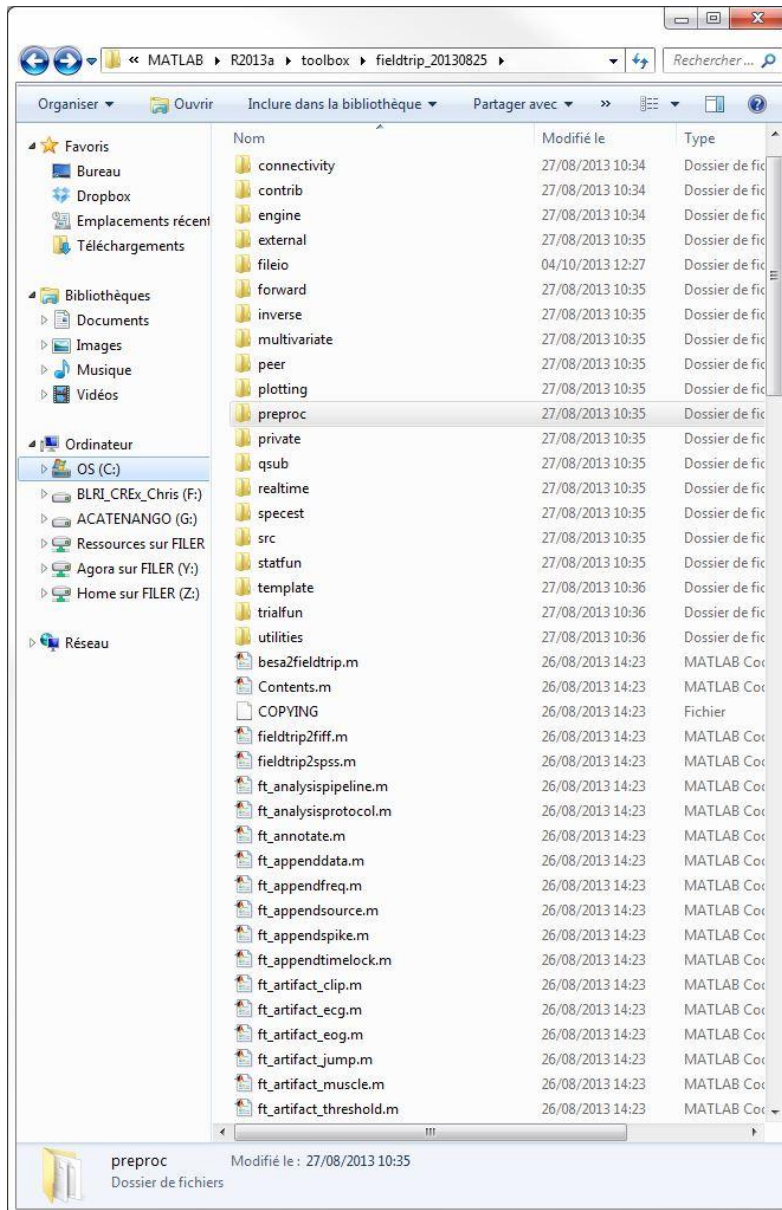
Niveau débutant en MEG et en statistiques.



- ➔ Une boîte à outil Matlab : des centaines de fonctions dédiées au traitement de données MEG
- ➔ Donders Institute (Pays-Bas) & collaborateurs

<http://fieldtrip.fcdonders.nl>

- ➔ Traitement des IRM anatomiques : lecture, segmentation
- ➔ Multiples choix de méthodes de calcul par fonctionnalité (ex. spectrogrammes) ; plusieurs algorithmes de localisation des sources



- Télécharger le dossier FieldTrip (une MAJ / jour)

- Ajouter le chemin d'accès aux chemins Matlab (Set path)

- Appeler les fonctions au niveau de la fenêtre de commande Matlab ou dans des scripts

Rappel - une fonction Matlab :  
`result = the_function(parameter_1, parameter_2,...)`

## Exemple 1 : extraire les données continues

➔ Utilisation de la fonction `ft_preprocessing`

```
cfg = [];  
cfg.dataset = 'F:\Catsem\S08\MEG\Run_3_Visu\c,rfhp0.1Hz';  
rawData = ft_preprocessing(cfg)
```

Les données  
extraites

La structure de  
paramètres (ici, un seul  
champ : dataset)

Chemin des  
données brutes

```
rawData = ft_preprocessing(cfg)
```

↓  
Structure : les données + les informations associées

rawData =

**hdr: [1x1 struct]**

➡ L'entête

**label: {276x1 cell}**

➡ Les noms des canaux  
(248 MEG + références + EOG + ECG + TRIGGER + RESPONSE)

**time: {[1 x 615194 double]}**

➡ Le vecteur temps (615194 points)

**trial: {[276 x 615194 double]}**

➡ Les mesures Nchan x Nsample

**fsample: 678.1700**

➡ La fréquence d'échantillonnage en Hz  
 $615194/678.17 = 907.1383 \text{ s} (\approx 15 \text{ min})$

**sampleinfo: [1 615194]**

➡ L'intervalle extrait  
*Tout le bloc de données continues*

**grad: [1x1 struct]**

➡ Infos sur les bobines & capteurs (positions, combinaisons)

**cfg: [1x1 struct]**

➡ Les paramètres utilisés par ft\_preprocessing

rawData =

hdr: [1x1 struct]

label: {276x1 cell}

time: {[1 x 615194 double]}

trial: {[276 x 615194 double]}

fsample: 678.1700

sampleinfo: [1 615194]

grad: [1x1 struct]

cfg: [1x1 struct]

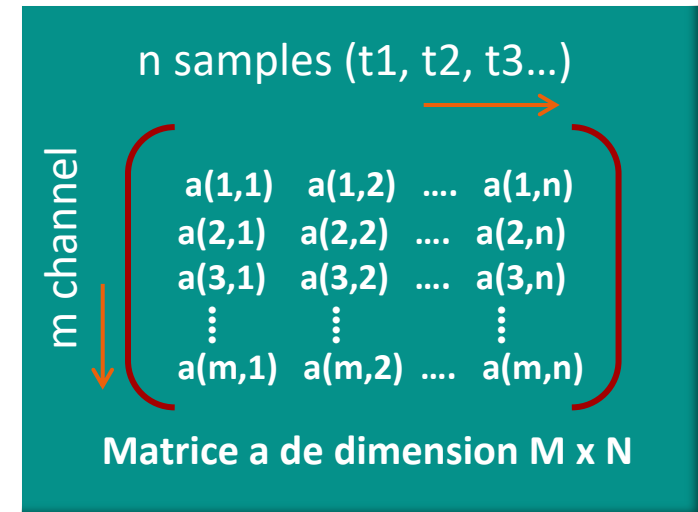
Un tableau nb\_chan x nb\_samples

- ▶ Données du capteur rangé en 3ème ligne :

```
rawData.trial{1}(3, :)
```

- ▶ Nom du capteur correspondant :

```
rawData.label{3}
```



## Exemple 2 : filtrer les données continues

➔ Toujours avec `ft_preprocessing`

>> `help ft_preprocessing`

The preprocessing options for the selected channels are specified with

`cfg.hpfilter` = 'no' or 'yes' highpass filter (default = 'no')

`cfg.hpfreq` = highpass frequency in Hz

`cfg.hpfilterord` = highpass filter order (default set in low-level function)

`cfg.hpfiltertype` = digital filter type, 'but' or 'fir' or 'firls' (**default = 'but'**)

Butterworth  
IIR filter

`cfg.hpfilterdir` = filter direction, 'twopass', 'onepass' or 'onepass-reverse' (**default = 'twopass'**)

**Valeur par défaut utilisée si non spécifiée**



## ➔ Exemple de filtrage :

Passe-haut à 0.5 Hz de type Butterworth ordre 2 sans déphasage

```
cfg = [];  
cfg.channel = { 'MEG' }; % Seulement les voies MEG  
cfg.hpfilter = 'yes'; % On demande un passe-haut  
cfg.hpfilterord = 2;  
cfg.hpfreq = 0.5;
```

```
filtData = ft_preprocessing(cfg, rawData);
```

Structure de  
paramètres

Structure de données  
obtenue lors de  
l'extraction

```
filtData = ft_preprocessing(cfg, rawData);
```



```
filtData =
```

```
    hdr: [1x1 struct]  
    fsample: 678.1700  
    grad: [1x1 struct]  
    sampleinfo: [1 615194]  
    trial: {[248 x 615194 double]}  
    time: {[1 x 615194 double]}  
    label: {248x1 cell}  
    cfg: [1x1 struct]
```

## ➡ Manipuler les données

Exemple : visualiser les données du capteur A142

```
>> plot(vecteur_X, vecteur_Y)
```

Trouve l'indice de la  
ligne correspondant au  
capteur A142  
(ichan = 197)



```
ichan = find(strcmp(rawData.label, 'A142')==1);
```

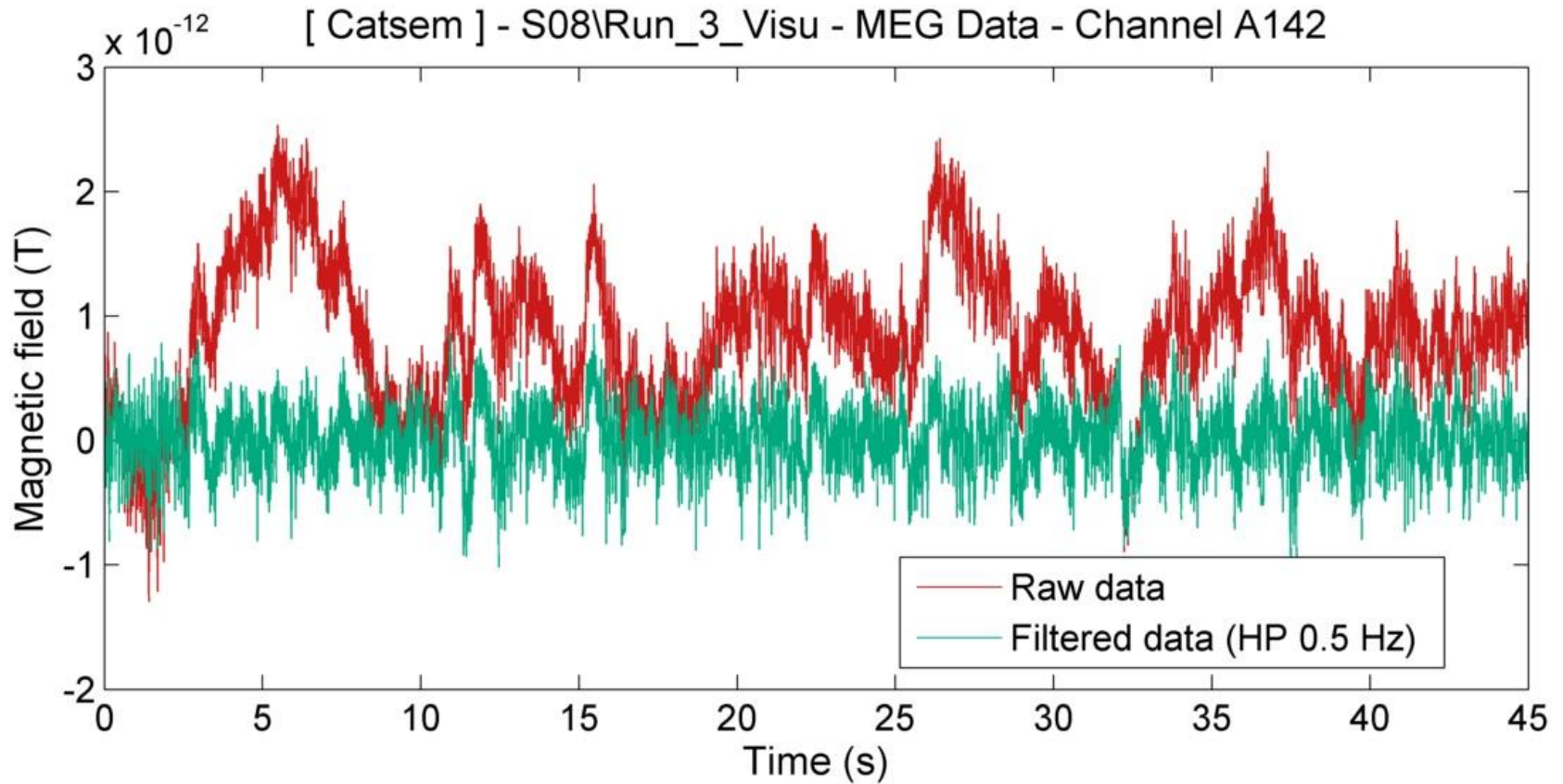
```
plot(rawData.time{1}, rawData.trial{1}(ichan,:))
```

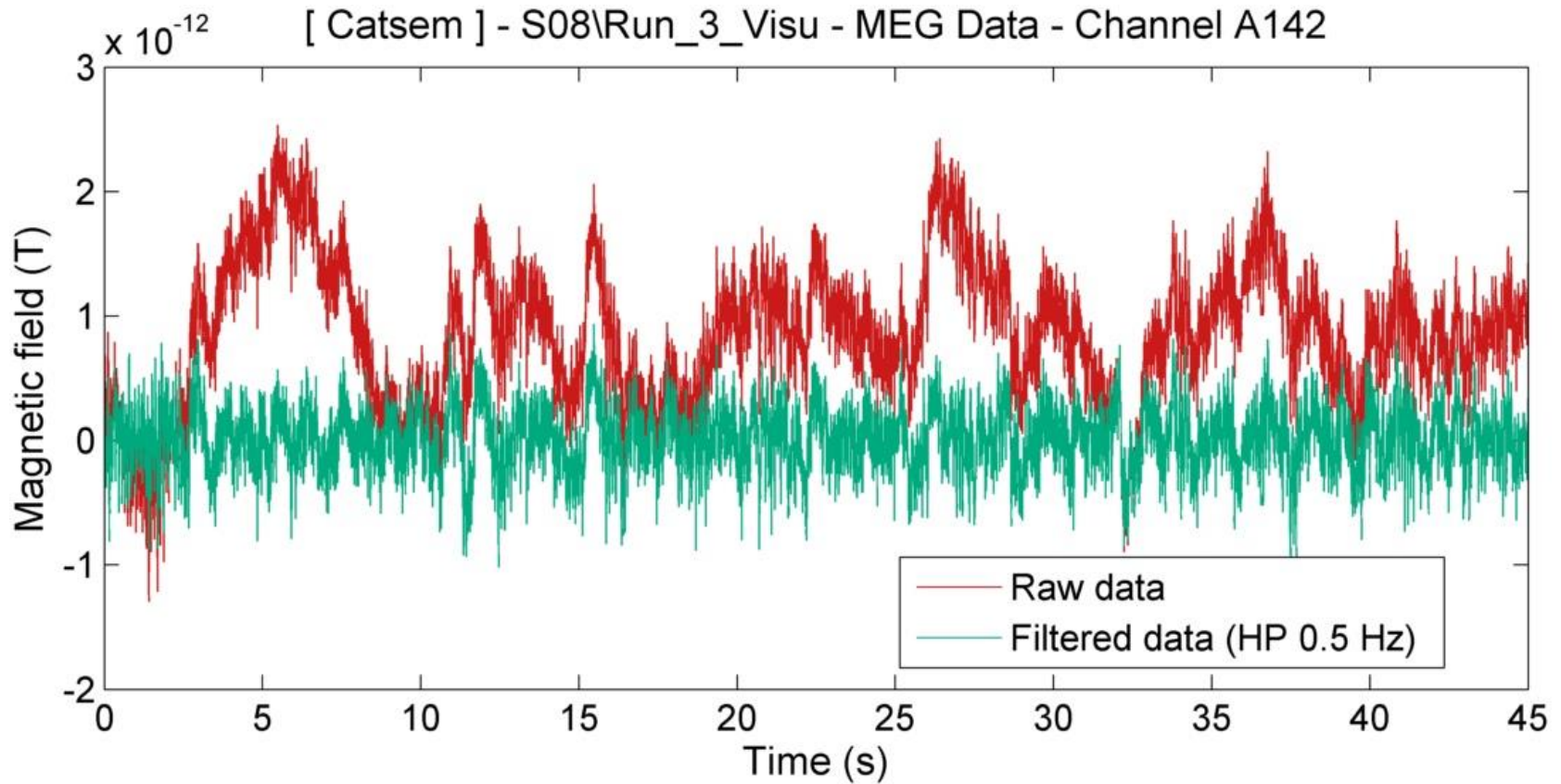
```
plot(filtData.time{1}, filtData.trial{1}(ichan,:))
```

```
xlim([0 45])
```

+ Mise en forme  
additionnelle







*ça semble avoir fonctionné...*

## Avantages

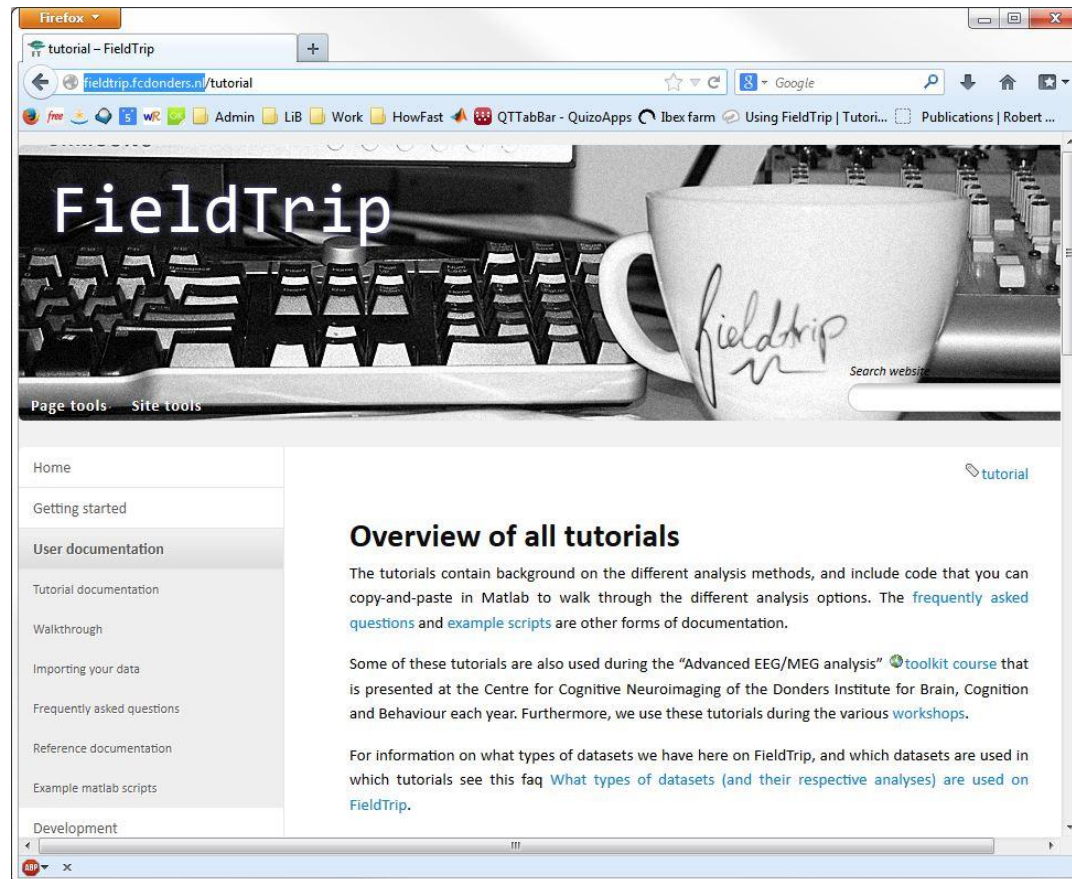
- Réalisation de scripts de traitement personnalisés, à partir des fonctions FieldTrip, Matlab et de ses propres fonctions
  - ↳ organiser ses étapes de traitement
  - ↳ inclure des calculs spécifiques (statistiques...)
  - ↳ mettre en forme des figures particulières
  - ↳ effectuer un traitement par lot, sur plusieurs jeux de données
- Données facilement manipulables
- Effet "boîte noire" limitée : traitement et paramètres choisis

## Inconvénients

- Tout ou presque est en ligne de commande (pas d'interface)
- Certaines fonctionnalités moins documentées, nécessité de chercher certains calculs dans les codes
- Affichage limité (en particulier pour la visualisation des données continues)

Pas d'interface avec des menus (Brainstorm, EEGLab)

➔ Pour savoir quelles fonctions utilisées : tutoriel indispensable



➔ Communauté très active – FieldTrip mailing list

Acquisition

Prétraitement

Nettoyage (artéfacts), filtrage, extraction des essais

Analyses spatio-temporelles

Champs évoqués et activité induite

Temps - Fréquence

Localisation des sources

Connectivité

Statistiques

Significativité, comparaison des conditions

**Pas de recette figée**

**Plusieurs méthodes de traitement possibles pour chaque type d'analyse**



Acquisition

Prétraitement

Analyses spatio-temporelles

Champs évoqués et activité induite

Localisation des sources

Connectivité

Statistiques



Exemples de  
traitements effectués à  
partir de FieldTrip pour  
les données du projet  
Catsem

**Données = signal cérébral d'intérêt + BRUIT**

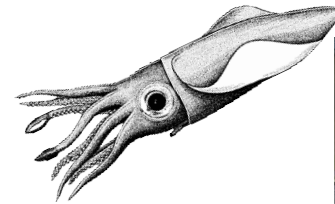
**Bruit minimisé au maximum :**

- ▶ Instrumentation optimisée : matériaux supraconducteurs maintenus à basse température
- ▶ Magnétomètres de référence soustraire le champ magnétique environnant et ses variations au champ enregistré au niveau du casque
- ▶ Chambre blindée faite de couches successives de métal ( $\mu$ -métal NiFe15Mo5, Cu, Al) permettant d'atténuer les perturbations HF liées aux ondes EM
- ▶ Préparation des sujets (retrait des éléments métalliques, maquillage...), consignes & entraînement

Données = signal cérébral d'intérêt + **BRUIT**

## Des artéfacts subsistent

- **Liés au système** : sauts de SQUID, capteur anormalement bruité, cassé ou saturé



- **Externes** : climatiseur, ascenseur, trafic proche, vibrations mécaniques transmises à la pièce



- **Internes** : activité physiologique (mouvements oculaires, cardiaques, musculaires)



- ▶ Nettoyer !
- ▶ Etapes de prétraitement proposées :

Filtrage pour retirer l'offset

Repérage & suppression des capteurs défectueux et des artéfacts énergétiques

Calcul des ICA, rejet des composantes indésirables

Extraction des essais (epoching) & tri

## Objectifs :

- Savoir repérer les données anormales
- Reconnaître des artéfacts
  - ↳ Choisir les composantes ICA à éliminer des données
  - ↳ Éliminer les mauvais essais

**Illustration avec les traitements réalisés à partir de FieldTrip et de scripts Matlab développés dans le cadre du projet Catsem**

Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

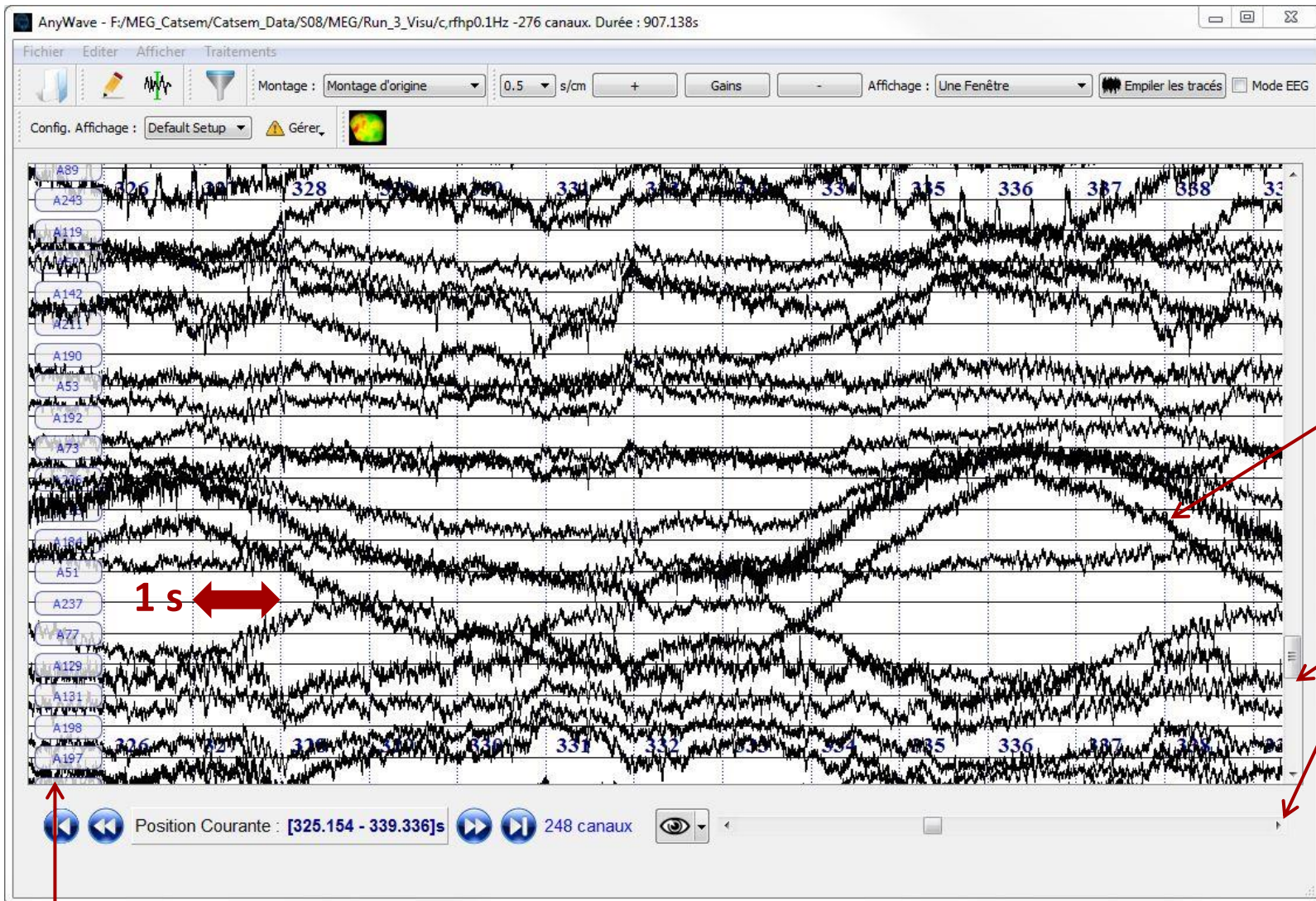
Nettoyage  
via l'ICA

Extraction et  
tri des essais

- Champ magnétique au cours du temps en Tesla ( $V \text{ s m}^{-2}$ )
- Données échantillonnées à 678 Hz (points/s)
- Pré-filtrées entre 0.1 et 200 Hz
- Blocs de 15 minutes

Une première visualisation des données continues (248 capteurs !)  
avec Anywave [*Centre MEG INS*] : navigation rapide & simplifiée



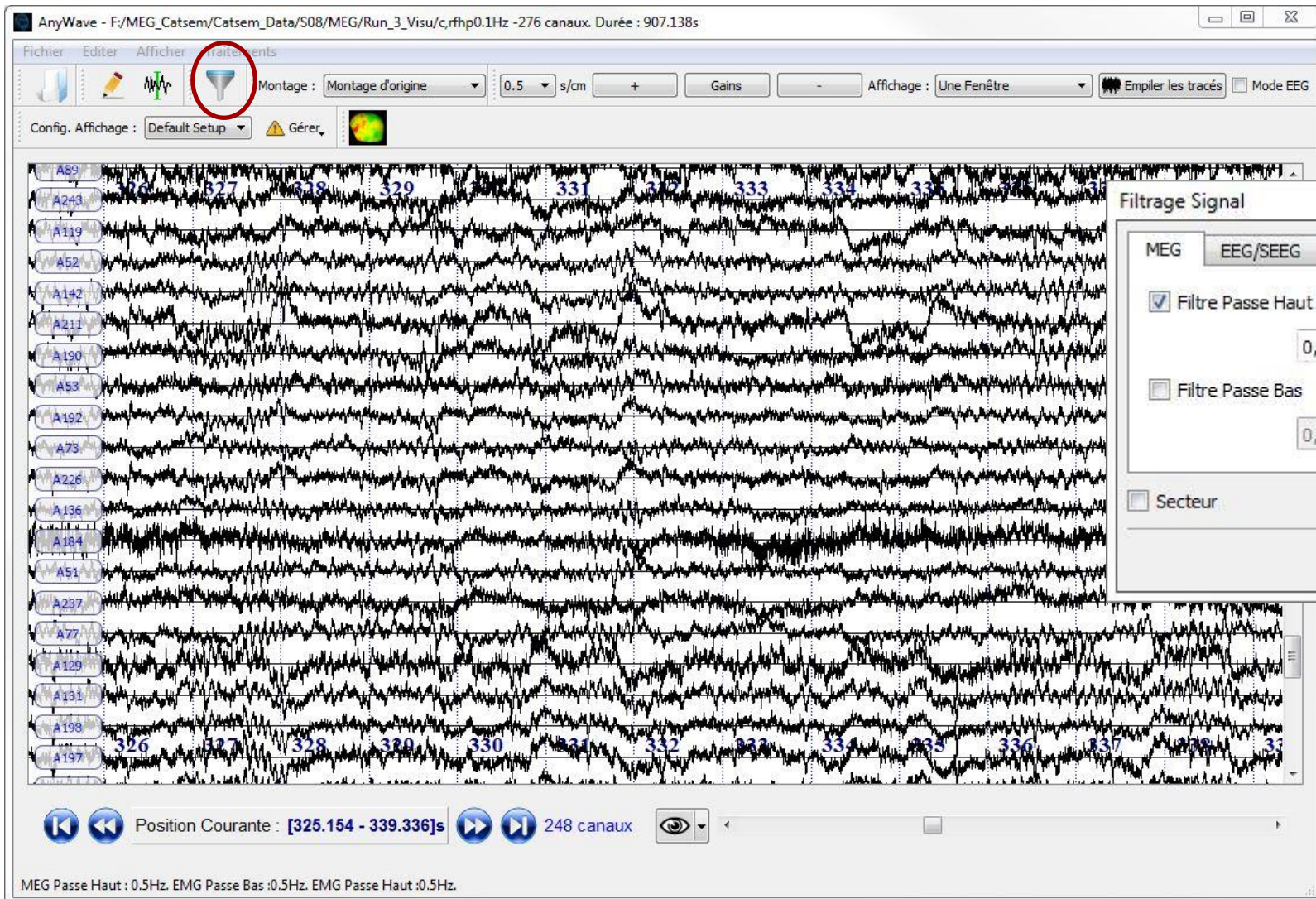


Variations  
BF 10 s

Barres de  
défilement

Noms des  
canaux





AnyWave - F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\c,rhfp0.1Hz - 276 canaux. Durée : 907.138s

Fichier Editer Afficher **Prétraitements**

Montage : Montage d'origine 0.5 s/cm Gains Affichage : Une Fenêtre Empiler les tracés Mode EEG

Config. Affichage : Default Setup Gérer

MEG EEG/SEEG EMG

Filtre Passe Haut 0,5 Hz

Filtre Passe Bas 0,1 Hz

Secteur

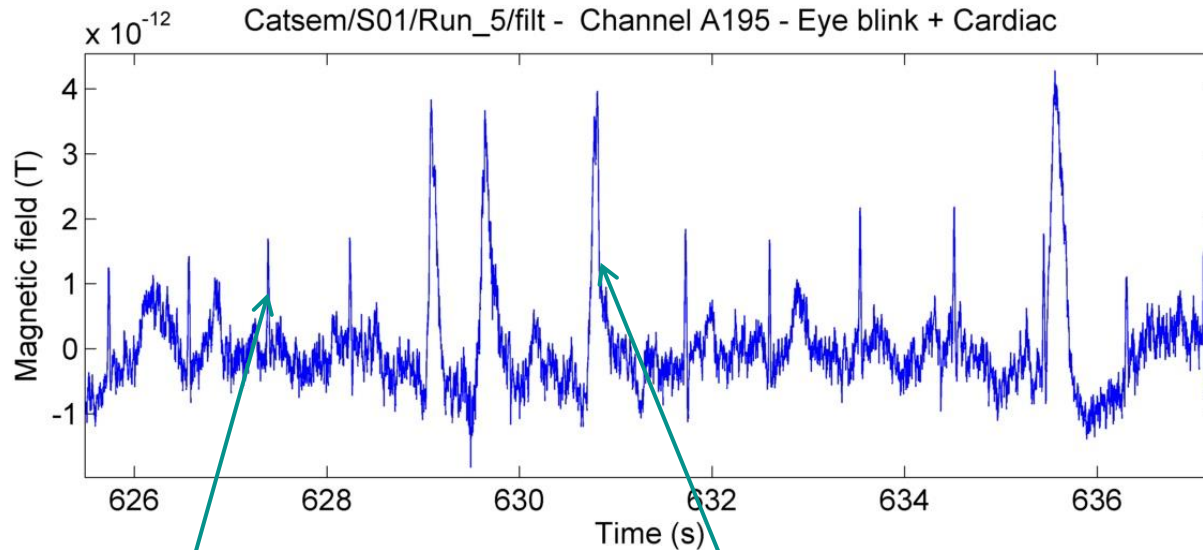
Appliquer

Position Courante : [325.154 - 339.336]s 248 canaux

MEG Passe Haut : 0.5Hz. EMG Passe Bas : 0.5Hz. EMG Passe Haut : 0.5Hz.

Tester l'effet du filtrage

## Allure des signaux visibles dans les données continues

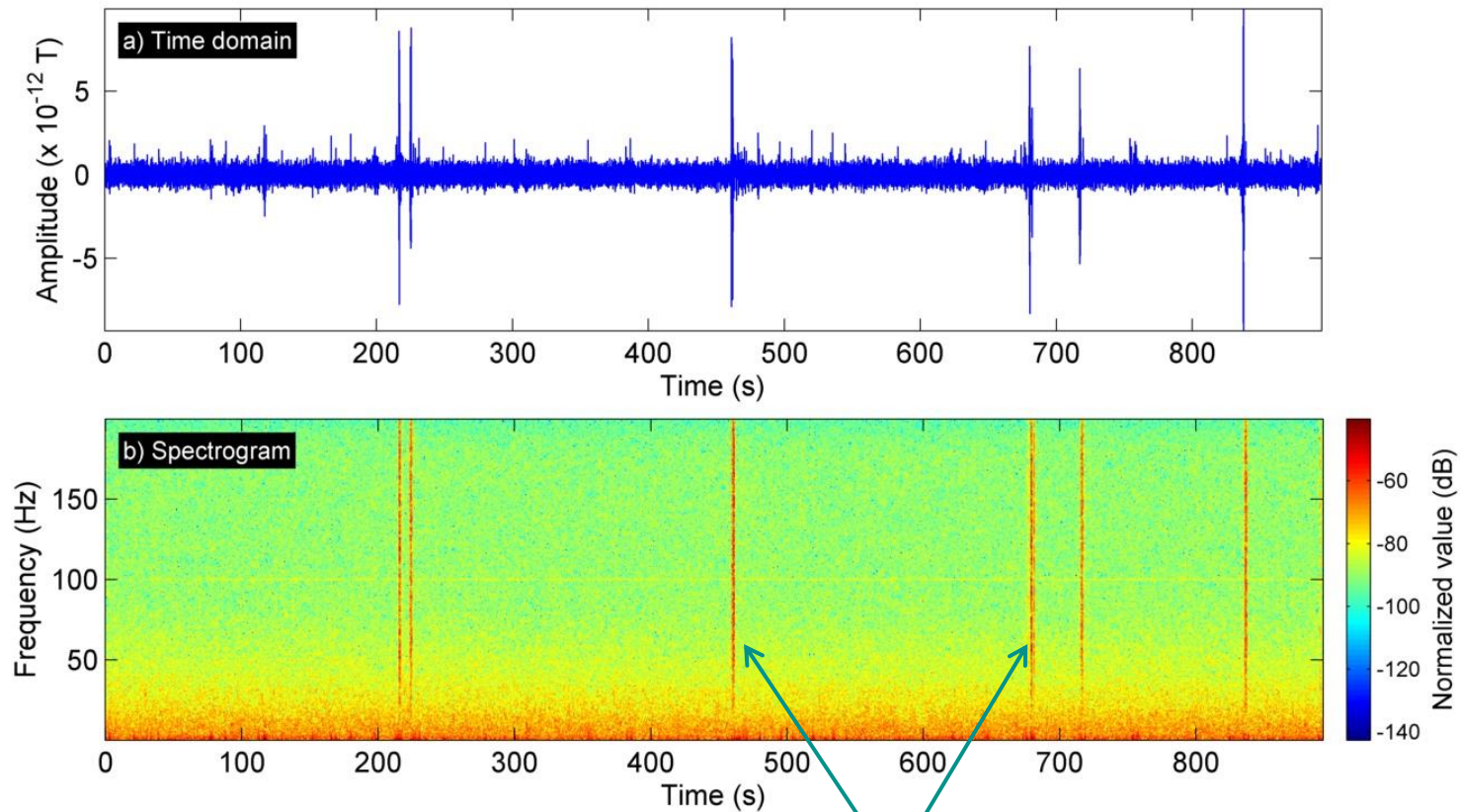


Lié aux pulsations cardiaques  
(très brefs et répétés toutes  
les 1 s)

Lié aux clignements (période  
de retour non fixe, entre 0.5  
et > 5s)

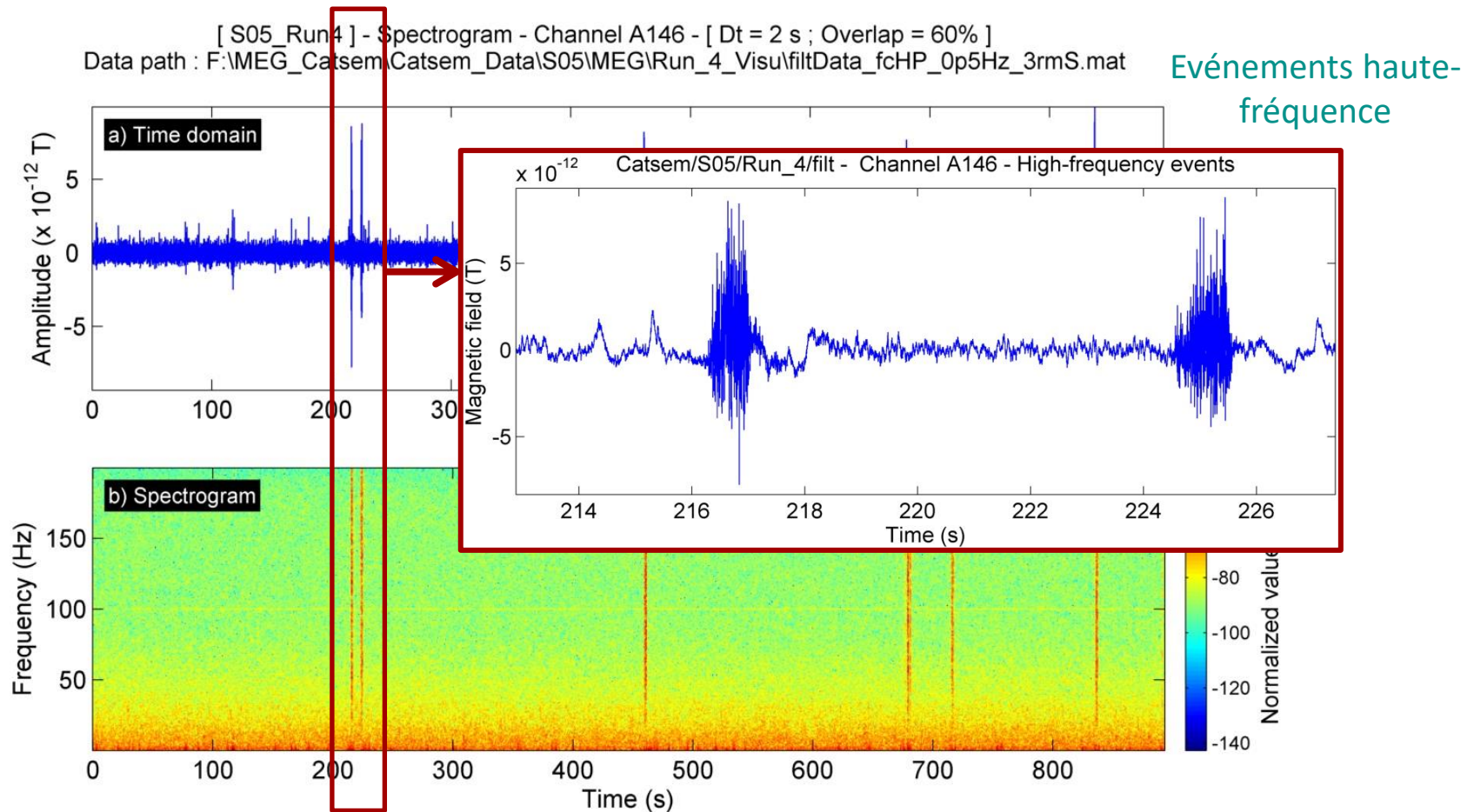
## Allure des signaux visibles dans les données continues

[ S05\_Run4 ] - Spectrogram - Channel A146 - [ Dt = 2 s ; Overlap = 60% ]  
Data path : F:\MEG\_Catsem\Catsem\_Data\S05\MEG\Run\_4\_Visu\filtData\_fcHP\_0p5Hz\_3rmS.mat



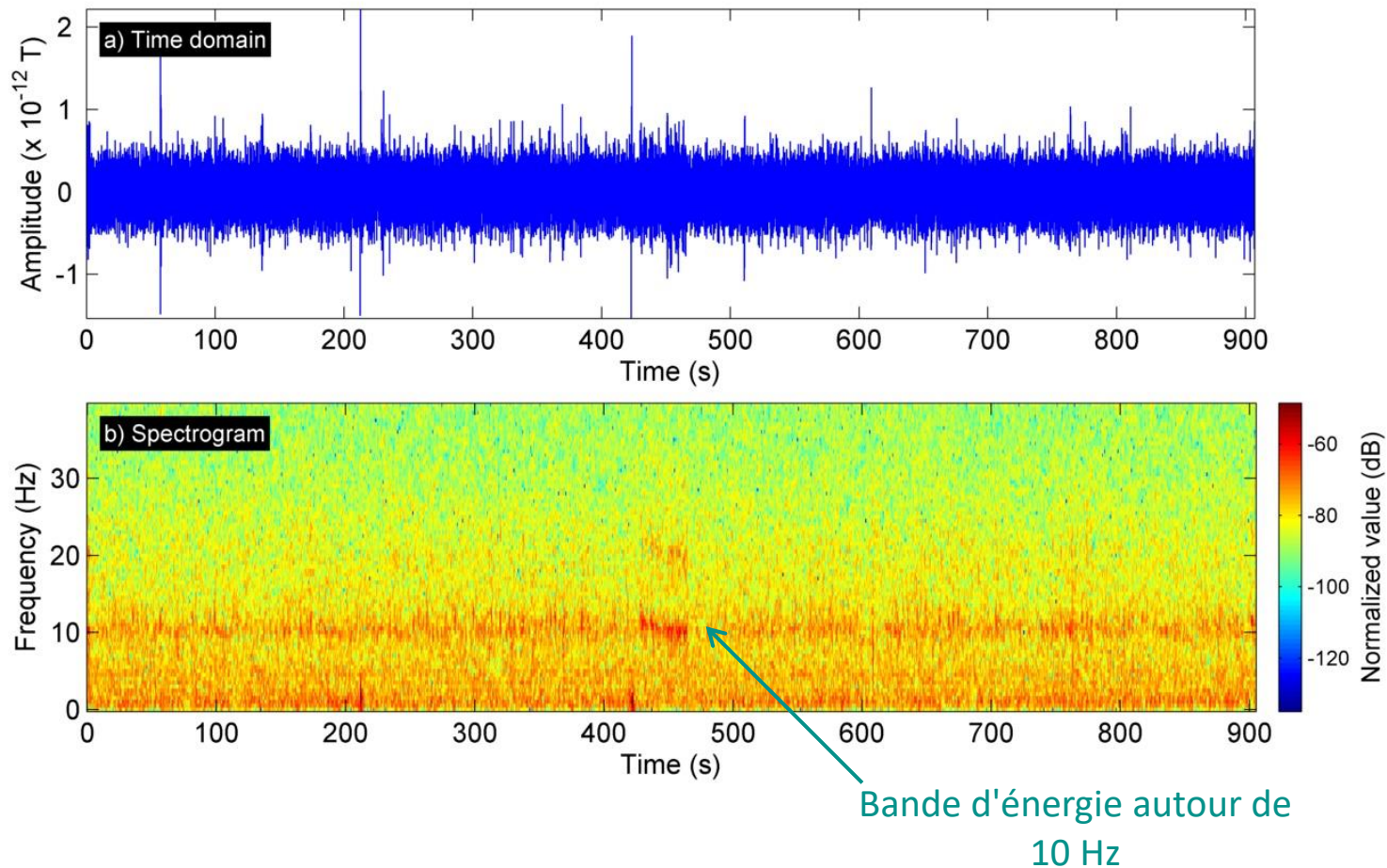
Événements brefs large-bande

## Allure des signaux visibles dans les données continues



## Allure des signaux visibles dans les données continues

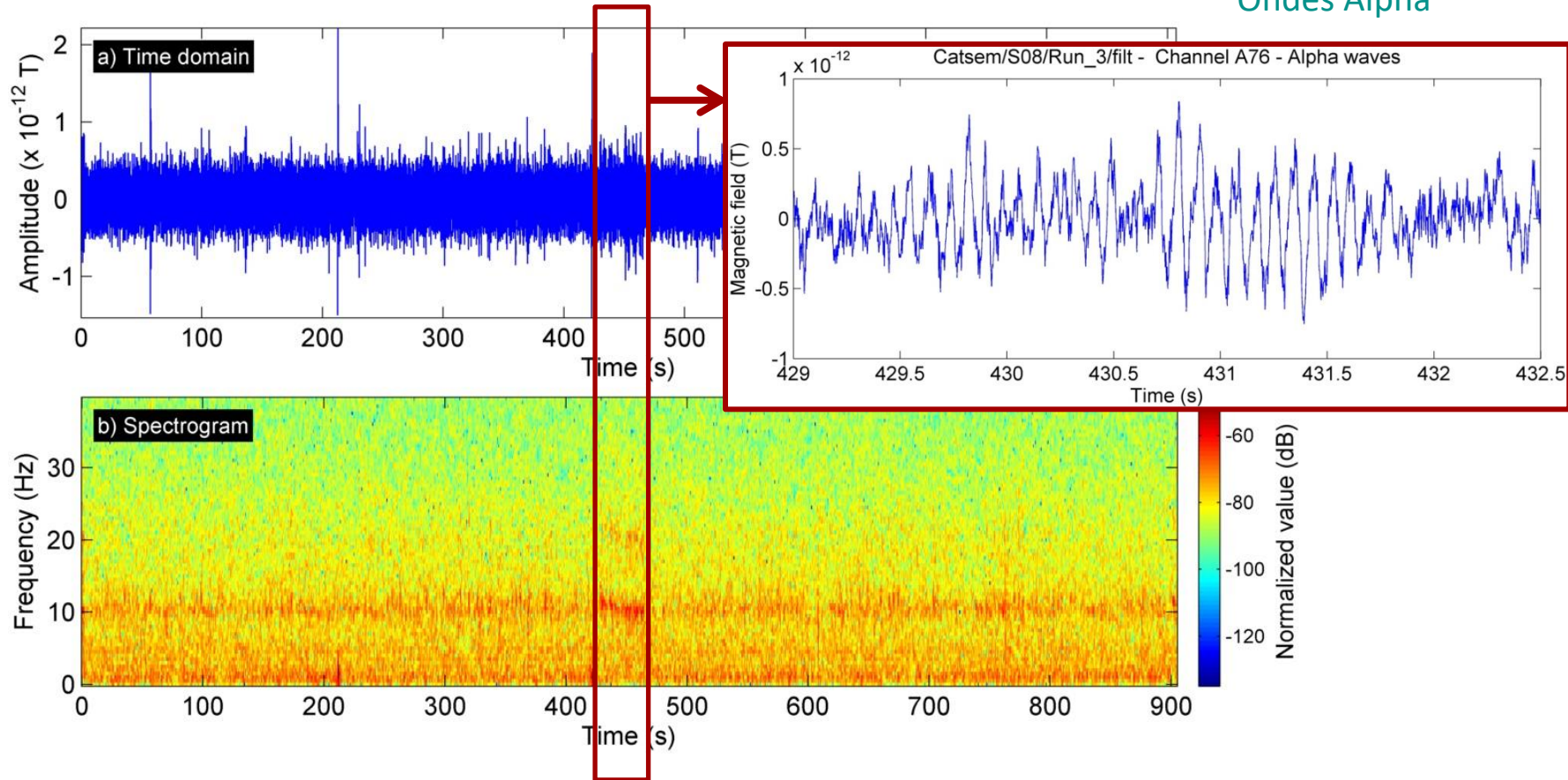
[ S08\_Run3 ] - Spectrogram - Channel A76 - [ Dt = 2 s ; Overlap = 60% ]  
Data path : F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\filtData\_fcHP\_0p5Hz.mat



## Allure des signaux visibles dans les données continues

[ S08\_Run3 ] - Spectrogram - Channel A76 - [ Dt = 2 s ; Overlap = 60% ]  
Data path : F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\filtData\_fcHP\_0p5Hz.mat

Ondes Alpha

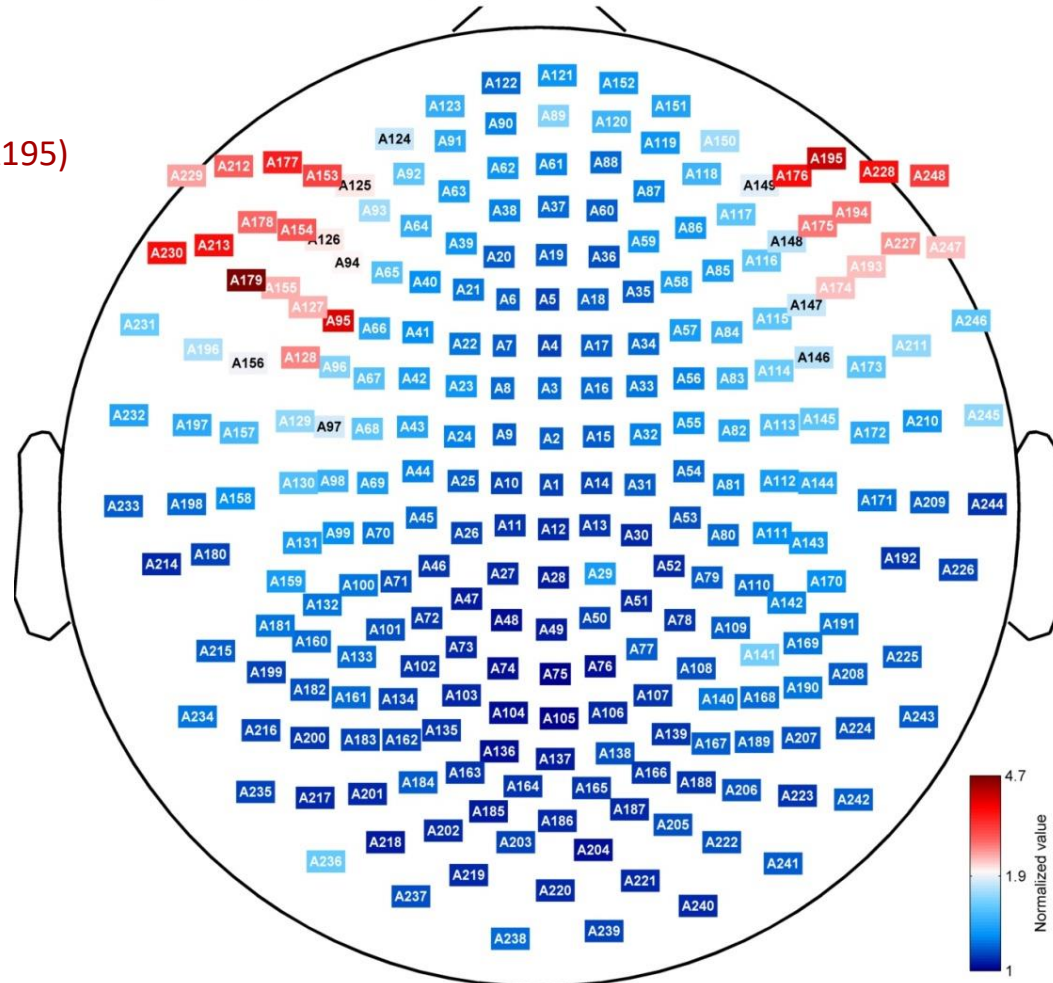


## Disposition des capteurs MEG – exemple de projection

F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\filtData\_fcHP\_0p5Hz  
Mean value of signal envelop (from hilbert transform)

Capteurs frontaux : signaux liés aux mouvements oculaires plus forts (A195)

Capteurs périphériques : plus sensibles aux pulsations



Capteurs centraux : ondes alpha plus marquées (A76)

Allure des signaux visibles dans les données continues

- ➔ A première vue : des artéfacts et des ondes alpha
- ➔ Ondes cérébrales en réponse au stimulus : trop faibles pour être distinguées
- ➔ Technique du **moyennage des essais (champs évoqués)** pour visualiser les **ondes synchrones** par rapport à la présentation du stimulus (la réponse visuelle précoce par exemple)
- ➔ **Moyennage des spectrogrammes** pour faire ressortir l'activité induite : réponses liées aux fonctions cognitives plus élevées, caractérisées par des **latences plus variables** suite à la présentation du stimulus

**Importance de l'étape de prétraitement pour assurer la qualité et la précision des résultats qui en découleront**



## Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

Double intérêt :

- Facilite l'inspection visuelle des données continues (suppression de l'offset) et des spectres
- Dans un second temps : centre les données avant le moyennage des essais (champs évoqués)

Catsem :

Variations BF importantes : passe-haut à 0.5 Hz sur les données continues

```
filtData = ft_preprocessing(cfg, rawData);
```

Filtrage

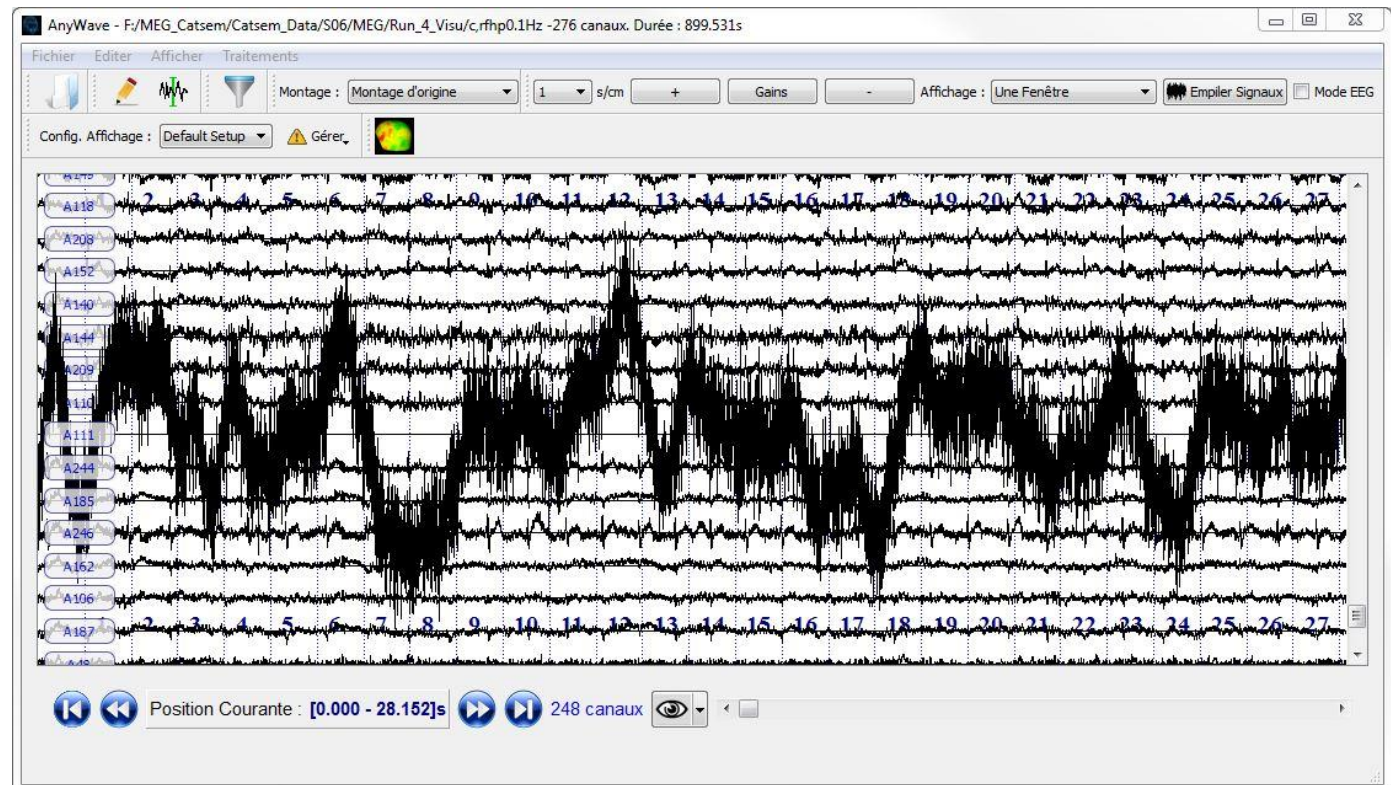
Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

- Inspection visuelle :
  - ↳ amplitude quasi-nulle
  - ↳ amplitude anormalement élevée
  - ↳ sauts d'amplitude anormaux



- Représentation dans le domaine de la fréquence (spectre)

Filtrage

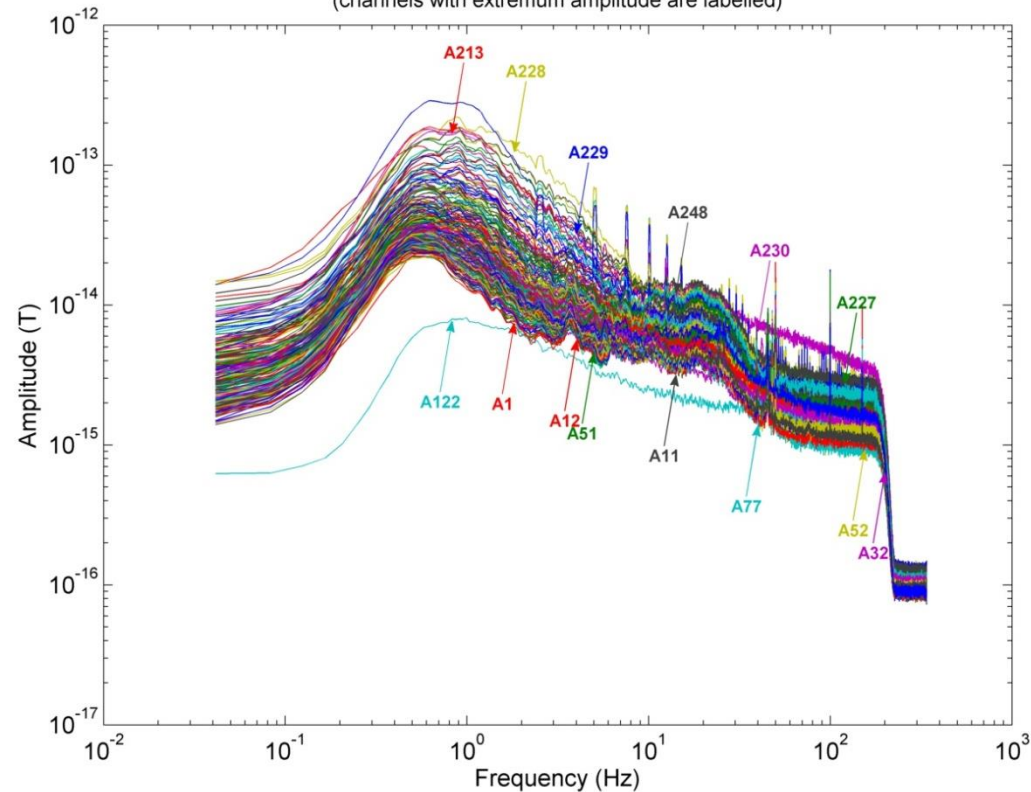
Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

[Smooth span=5] Stacked spectrum of MEG filtered data - 246 channels - [ nsp = 30 ; lgsp = 20 s ; ti = 40 s ]  
 datapath : F:\MEG\_Catsem\Catsem\_Data\S01\MEG\Run\_5\_Visu\filtData\_2rma\_fcHP\_0p5Hz.mat  
 (channels with extremum amplitude are labelled)



➔ Repérage immédiat des capteurs anormaux (A122, A230)

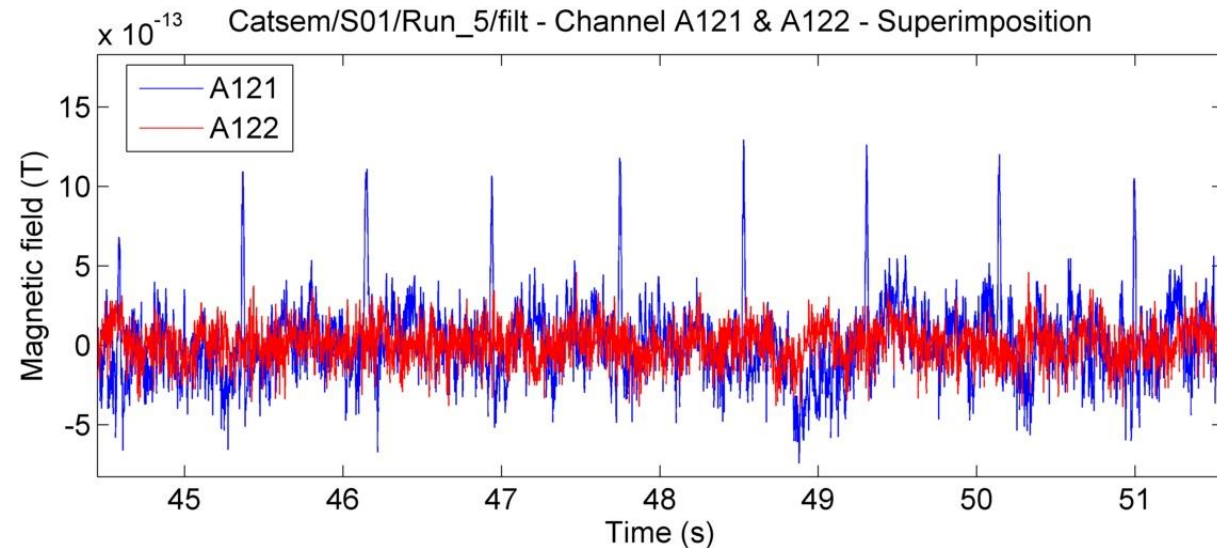
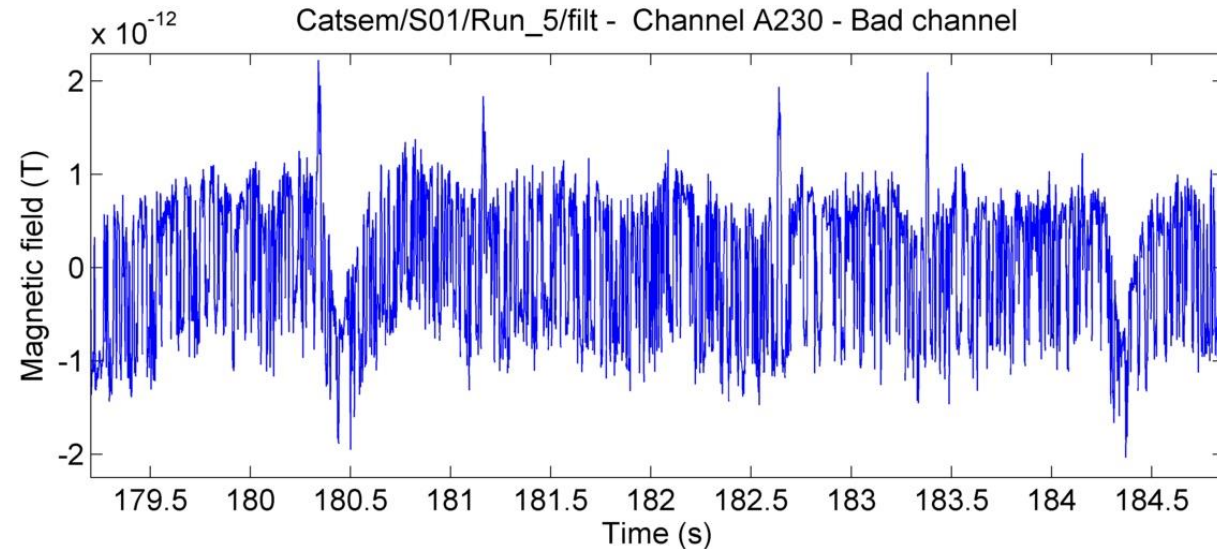
Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

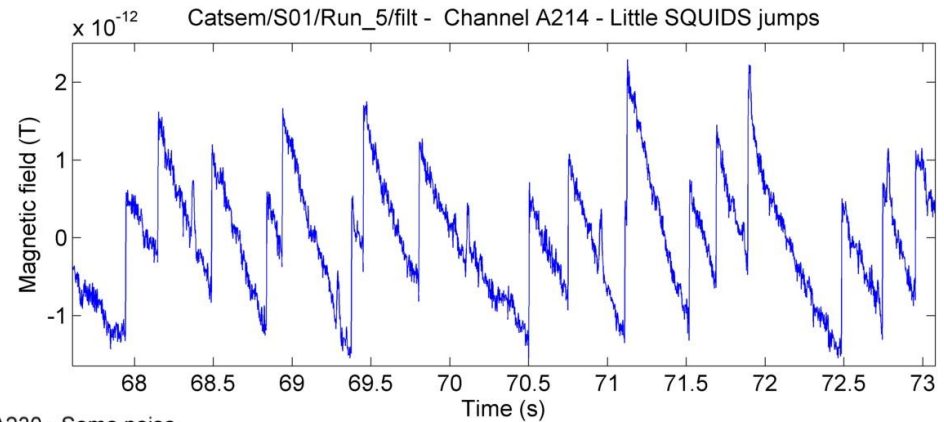
Nettoyage  
via l'ICA

Extraction et  
tri des essais



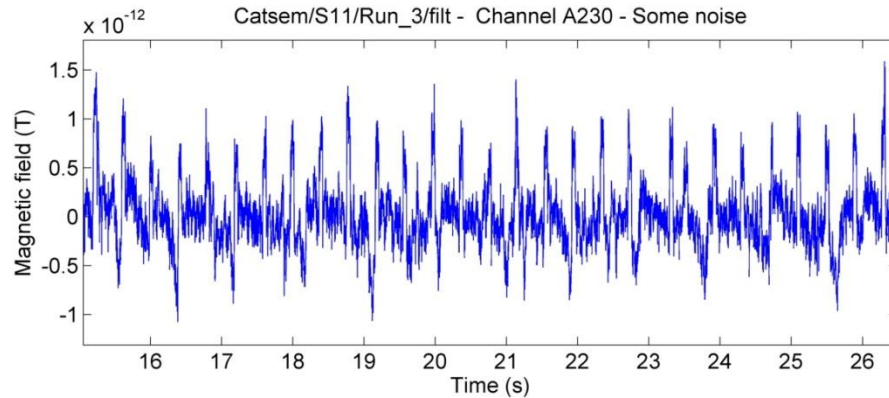
Filtrage

Exemple d'autres types de bruit repérés

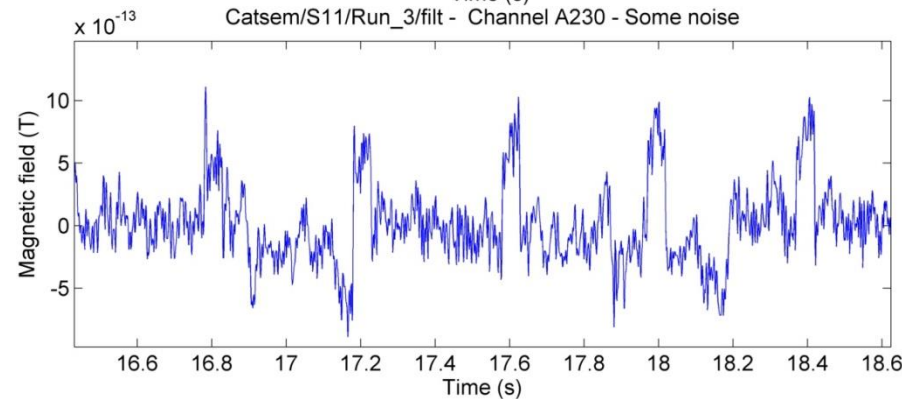


Capteurs défectueux

Artéfacts énergétiques



Nettoyage via l'ICA



Extraction et tri des essais

Filtrage

- Les capteurs présentant des données très abîmées sont supprimées du jeu de données

Capteurs  
défectueux

Exemple pur supprimer les capteurs A111 et A212 :

```
cfg = [];  
cfg.channel = {'MEG', '-A111', '-A212'};  
rawDataOK = ft_preprocessing(cfg, rawData);
```

Artéfacts  
énergétiques

- Certains artéfacts répétés de faible amplitude pourront être atténués en rejetant la composante ICA associée

Nettoyage  
via l'ICA

- Des méthodes de détection automatique existent : seuils à déterminer, vérifications indispensables des résultats

Extraction et  
tri des essais

Filtrage

Capteurs  
défectueux

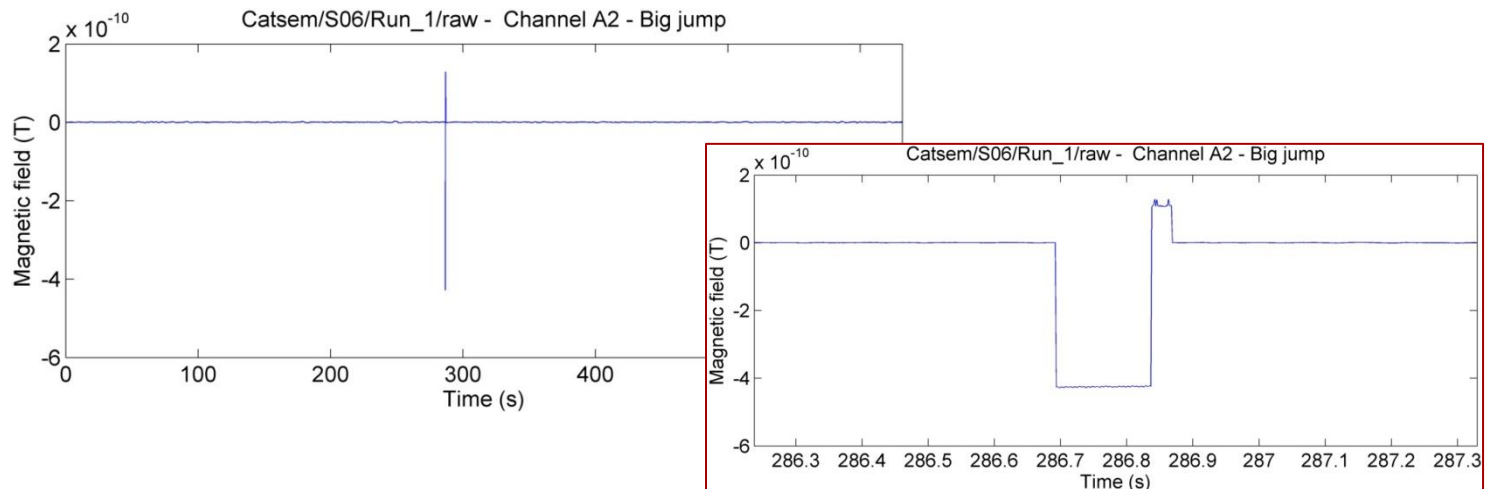
Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

- Très énergétiques (visibles sur tous les canaux) : pics d'amplitudes très élevée et de courte durée, correspondant principalement à des sauts de SQUIDS
- A ôter impérativement
  - ↳ perturbent les représentations fréquentielles
  - ↳ empêchent l'utilisation des ICA

## Exemple de signal au niveau d'un capteur



## Exemple : données au niveau de 6 capteurs et spectres associés

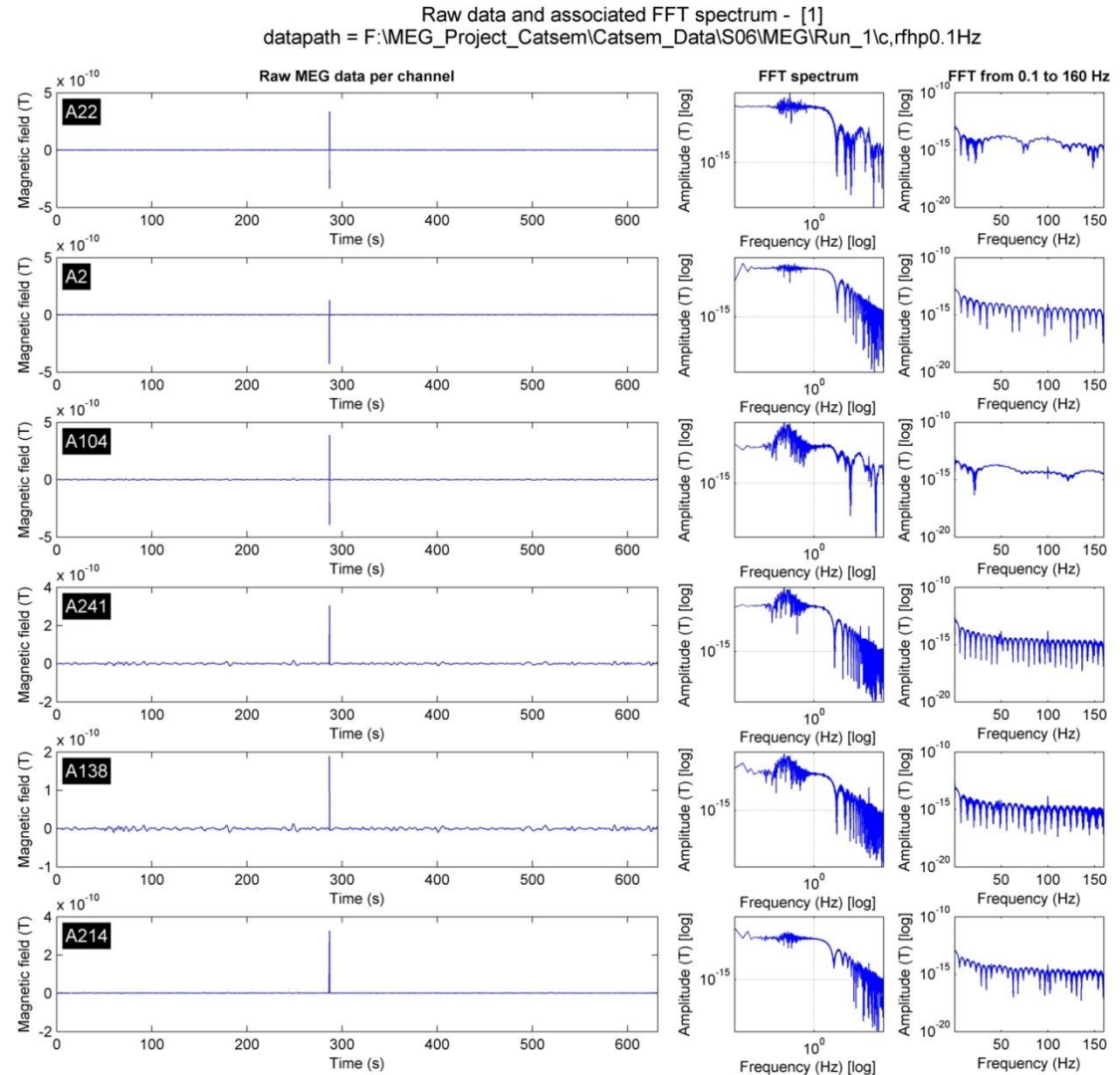
Filtrage

Capteurs défectueux

Artéfacts énergétiques

Nettoyage via l'ICA

Extraction et tri des essais





Filtrage

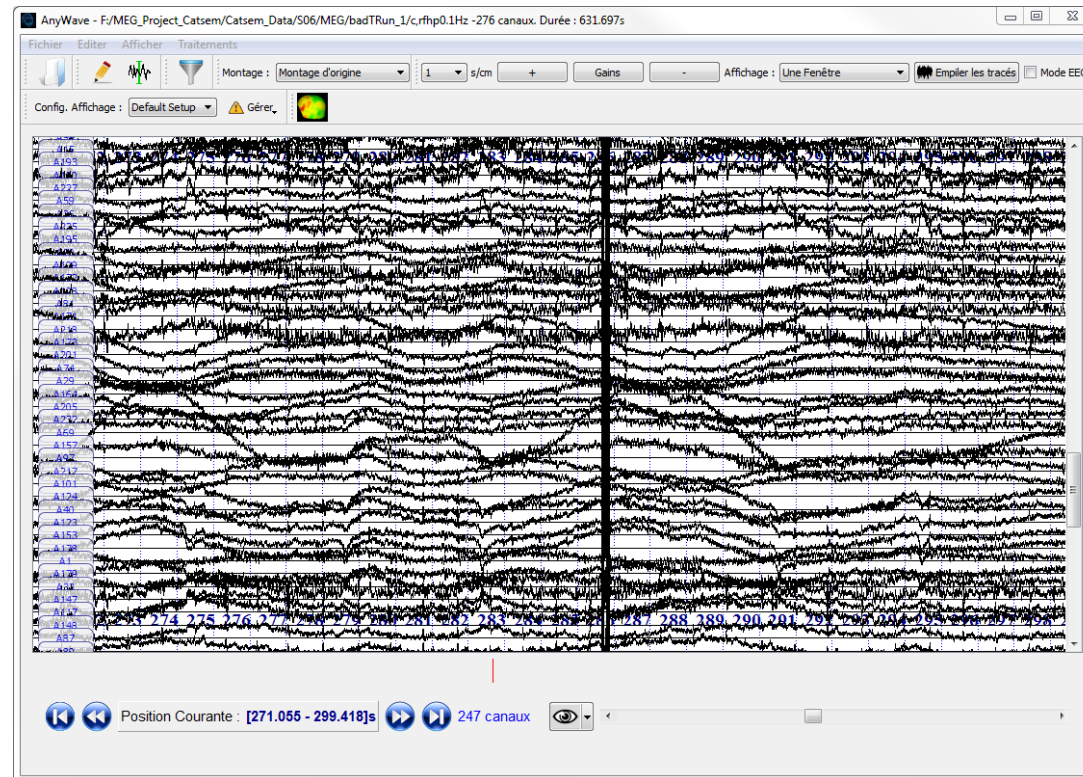
Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

Exemple via Anywave : "un mur"



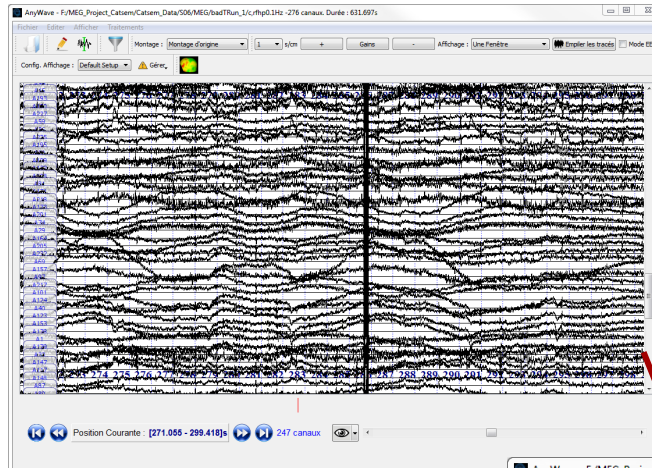
Filtrage

Capteurs  
défectueux

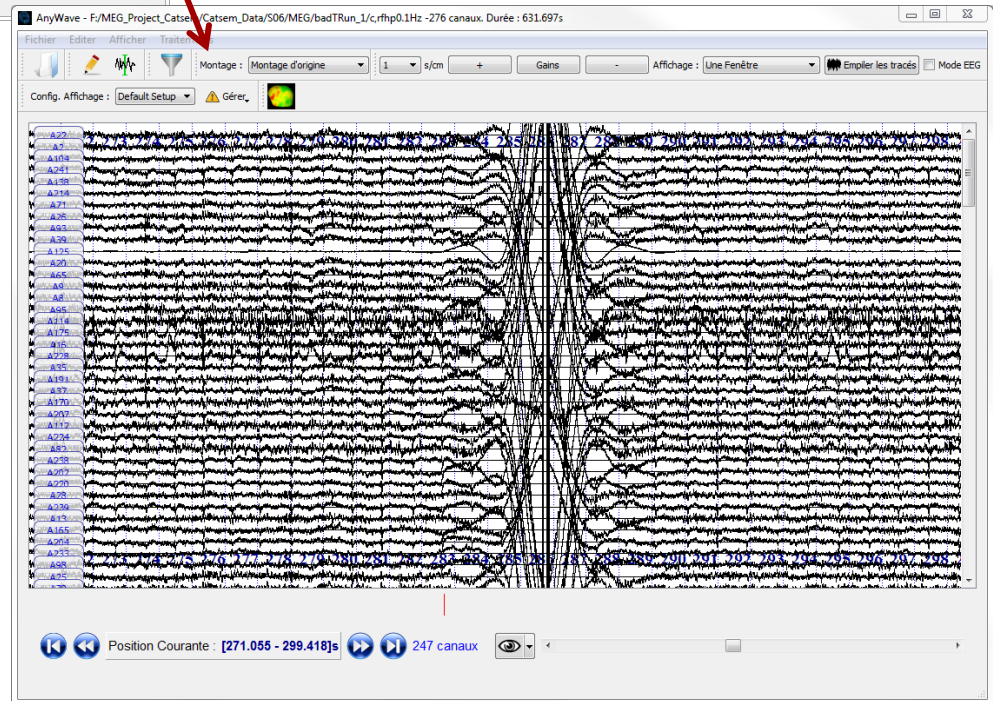
Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais



Effet du filtrage



Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

- Méthode de suppression :
  - ▶ Pour chaque capteur, la partie des données impactées par le pic d'amplitude est remplacée par la portion de données de même durée qui le précède
  - ▶ Données filtrées (pas de fluctuations BF)
  - ▶ Les triggers indiquant la présence de stimulus pendant l'intervalle du pic sont supprimés

```
wina=[283 290];  
% Eliminate artefacts defined by wina  
padData =  
meg_padding_artefact(filtData, wina);
```

Filtrage

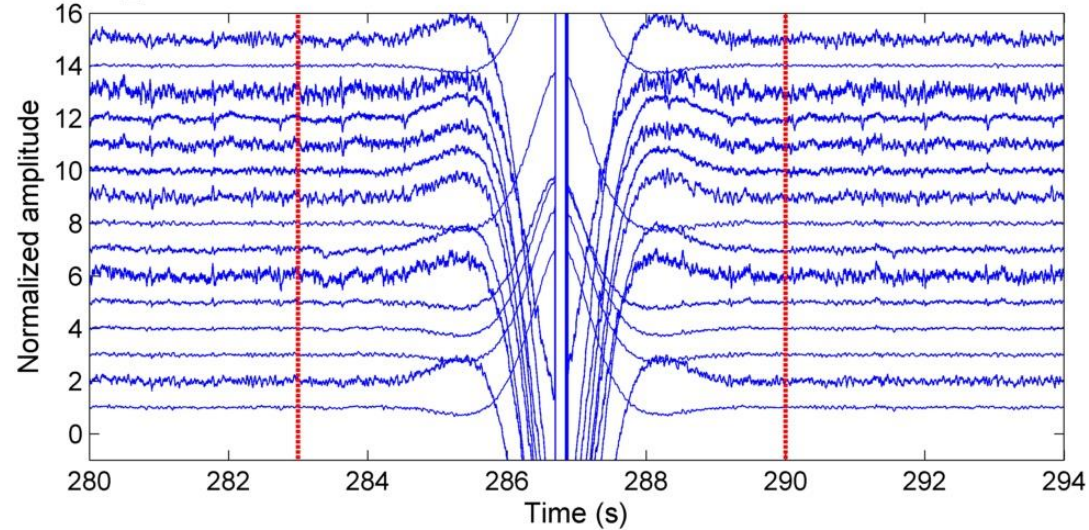
Capteurs  
défectueux

Artéfacts  
énergétiques

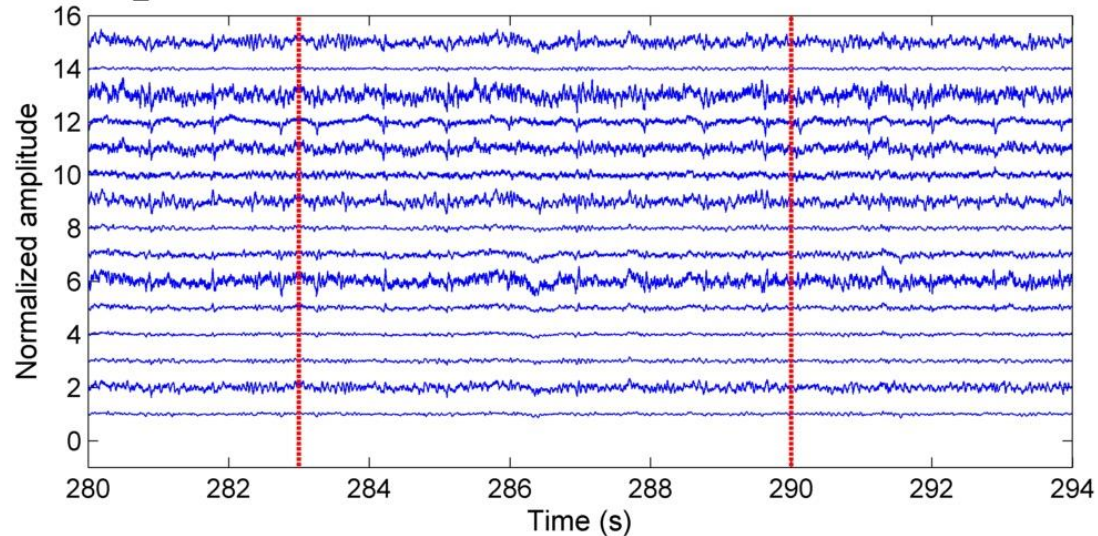
Nettoyage  
via l'ICA

Extraction et  
tri des essais

S06/Run\_1 - First 15 channels in the filtered data set around the artefact - BEFORE its removal



S06/Run\_1 - First 15 channels in the filtered data set around the artefact - AFTER its removal



Filtrage

▶ Données prêtes pour l'analyse en composantes indépendantes (*Independent Component Analysis*)

Capteurs défectueux

▶ Extraction des composantes linéaires aussi indépendantes que possibles

Artéfacts énergétiques

• Traitement de signaux multi-capteurs à l'écoute d'un ensemble discret de sources de signal (ex. acoustique)

Nettoyage via l'ICA

• Ces sources sont supposées statistiquement indépendantes (pour des raisons physiques)

Extraction et tri des essais

• Chaque composante est monodimensionnelle et le nombre de composantes correspond au nombre de dimension des mesures (capteurs)

Filtrage

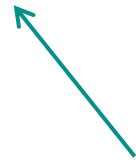
Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

```
cfg = [];  
cfg.demean = 'no'; % Default = 'yes'  
cfg.method = 'runica'; % EEGLab  
ICAcomp = ft_componentanalysis(cfg, padData);
```



ICAcomp =

```
fsample: 678.1700  
time: {[1x603268 double]}  
trial: {[248x603268 double]}  
topo: [248x248 double]  
unmixing: [248x248 double]  
label: {248x1 cell}  
topolabel: {248x1 cell}  
grad: [1x1 struct]  
sampleinfo: [1 603268]  
cfg: [1x1 struct]
```

Filtrage

Capteurs  
défectueux

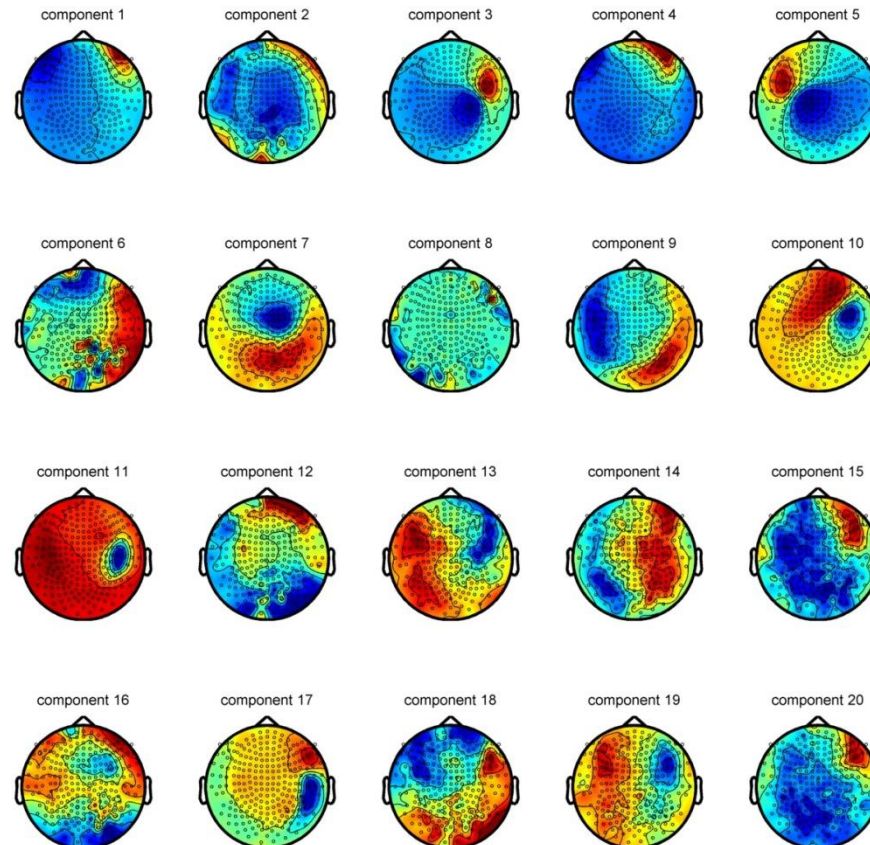
Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

Exemple pour un jeu de données : topographies des 20 premières composantes

ICA from RUNICA calculation - Topographic plot [1]  
datapath = F:\MEG\_Project\_Catsem\Catsem\_Data\S07\MEG\Run\_3\ICAcomp\_preprocData\_fcHP\_0p5Hz.mat



Pour aider au choix des composantes liées aux artéfacts :  
visualisation des topo + décours temporel

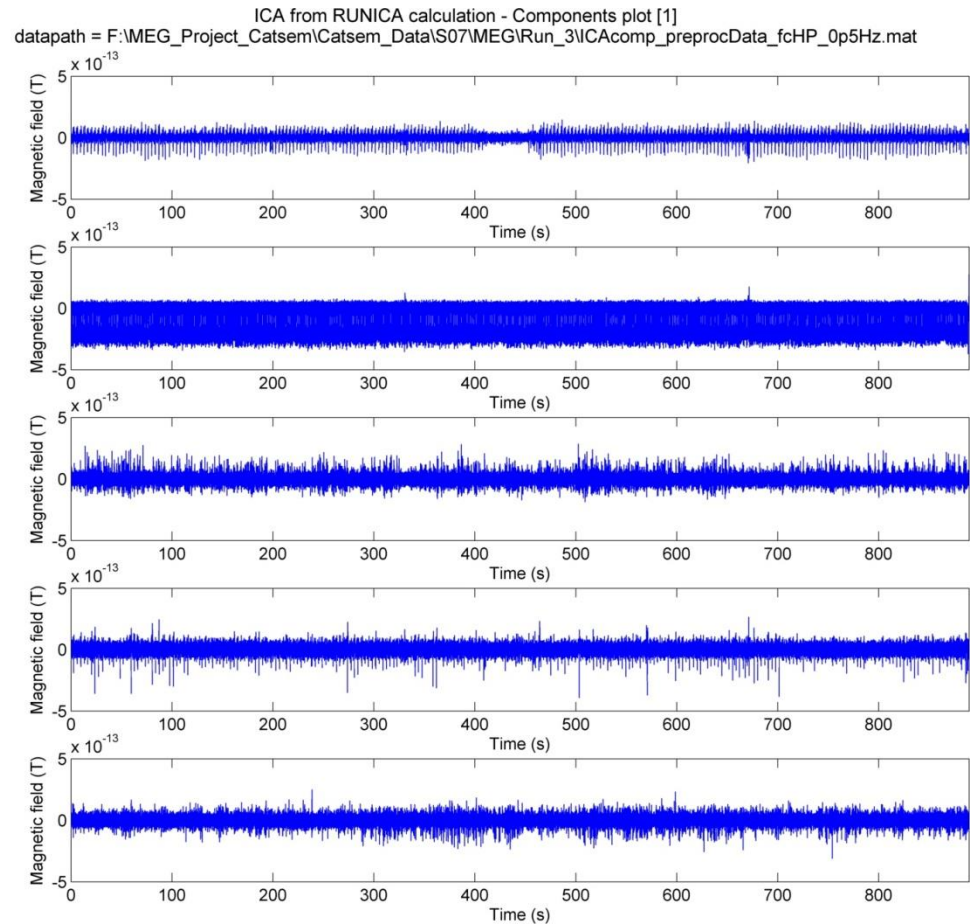
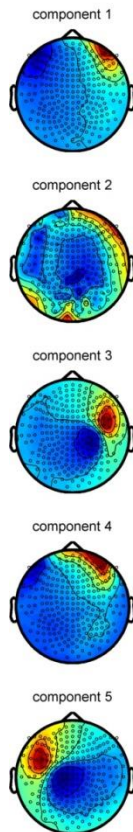
Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais





## Composantes 1 à 5

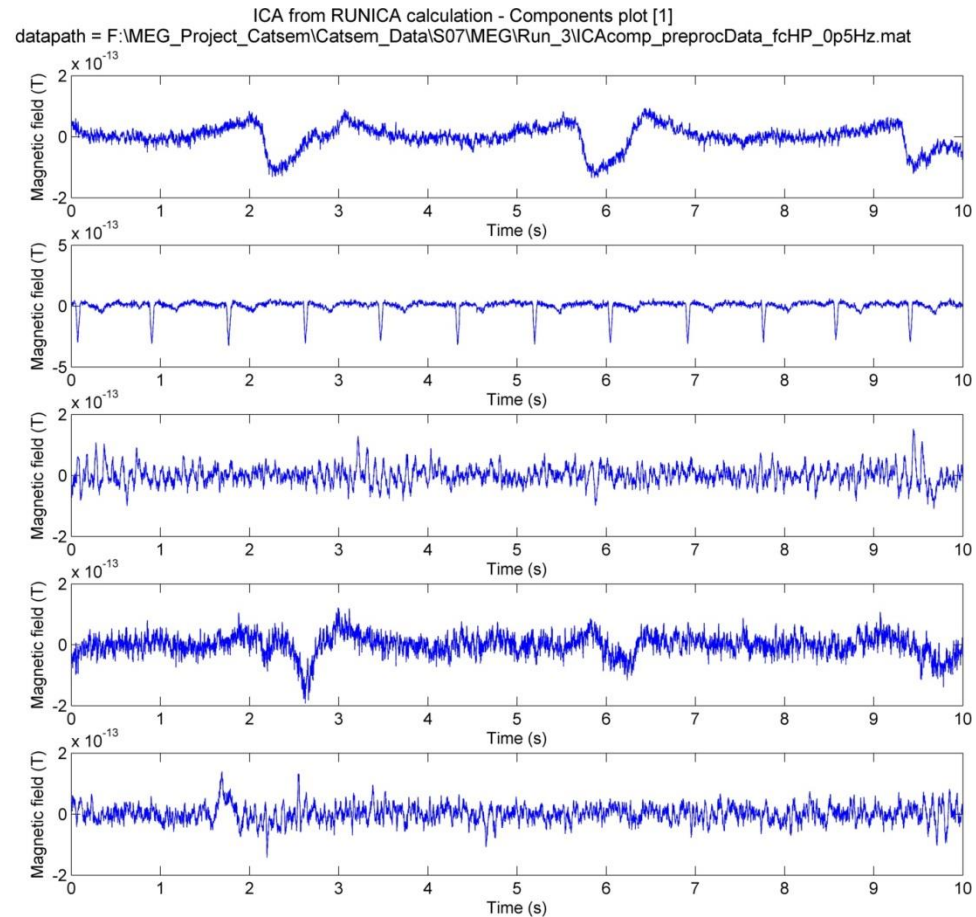
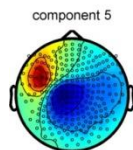
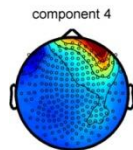
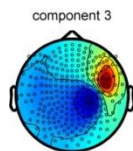
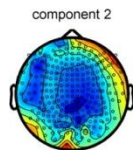
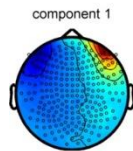
Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais



Oculaire  
(frontal)

Cardiaque  
(anneau)

Oculaire  
(frontal)

## Composantes 6 à 10

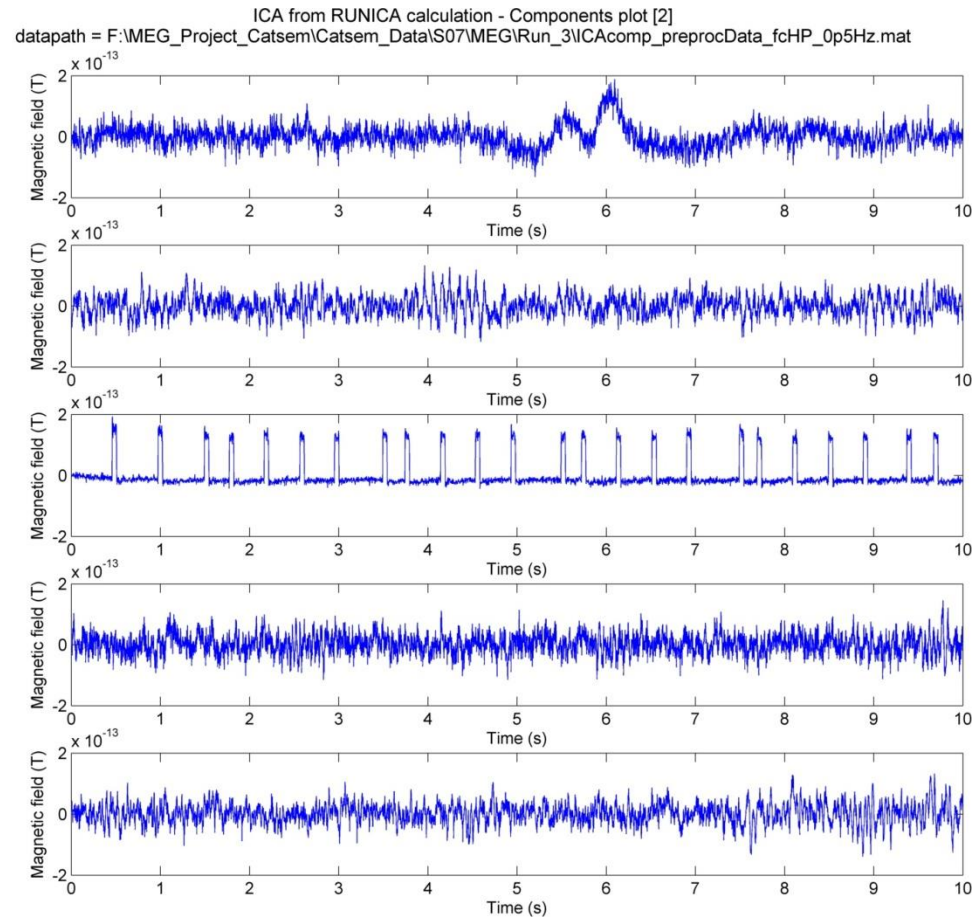
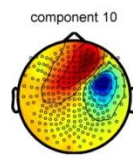
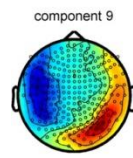
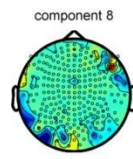
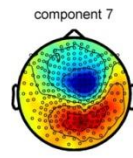
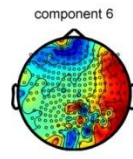
Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais



Indéterminé

Alpha

Signaux carrés  
(électronique)

Filtrage

- ▶ Composantes à repérer : liées aux artéfacts physiologiques reconnus : mouvements oculaires et cardiaques (+ ici : signaux carrés)

Capteurs défectueux

- ▶ Seulement ce qui est identifiable ! (et que l'on a repéré dans les données)

Artéfacts énergétiques

Nettoyage via l'ICA

```
cfg = [];  
cfg.component = [1 2 4 8];  
cleanData = ft_rejectcomponent(cfg, ICAcomp, padData);
```

Extraction et tri des essais



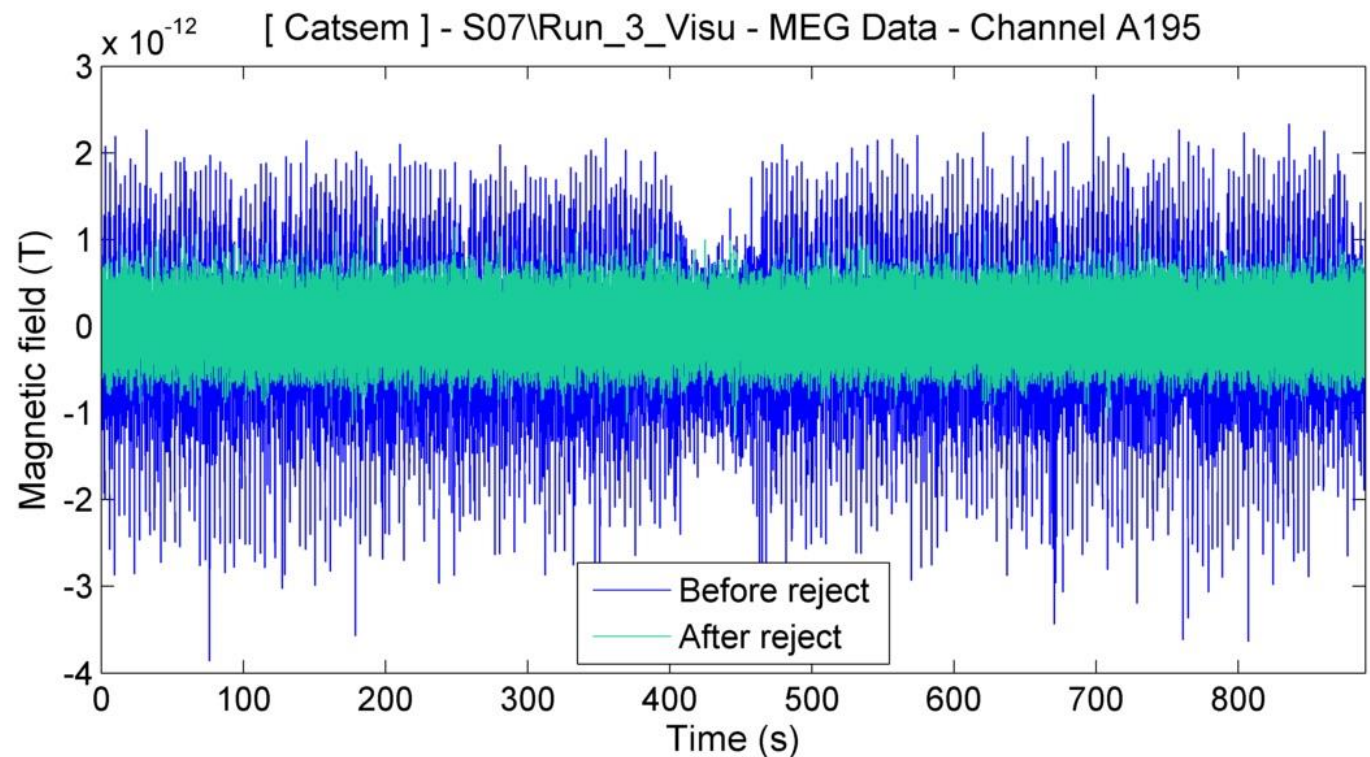
Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais



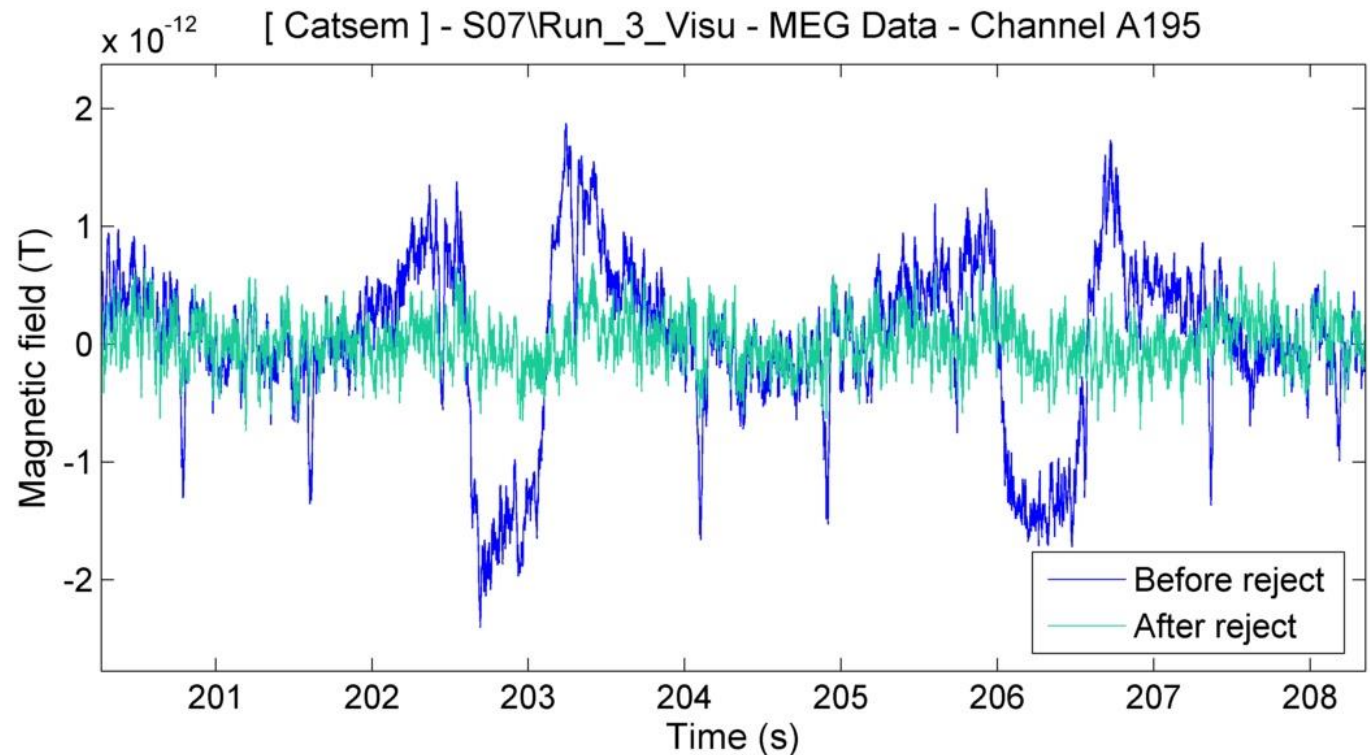
Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais



Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

- ▶ Extraction des portions de données autour des triggers des stimuli visuels (ex entre -500 et 1000 ms)
- ▶ Code enregistré dans la voie "TRIGGER" des données spécifique à la catégorie de mot présenté
- ▶ 60 essais par type de mot (par condition) : régulier haute-fréquence [536], régulier basse-fréquence [534] irrégulier haute-fréquence, irrégulier basse-fréquence [532] et corps [592]

```
ft_definetrial & ft_redefinetrial
```

Filtrage

- ▶ Certains artéfacts n'ont pas pu être repéré ou nettoyé (ex. événements HF dus aux mouvements du sujet)

Capteurs  
défectueux

- ▶ Identifier les essais trop bruités afin de les supprimer du jeu de données

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

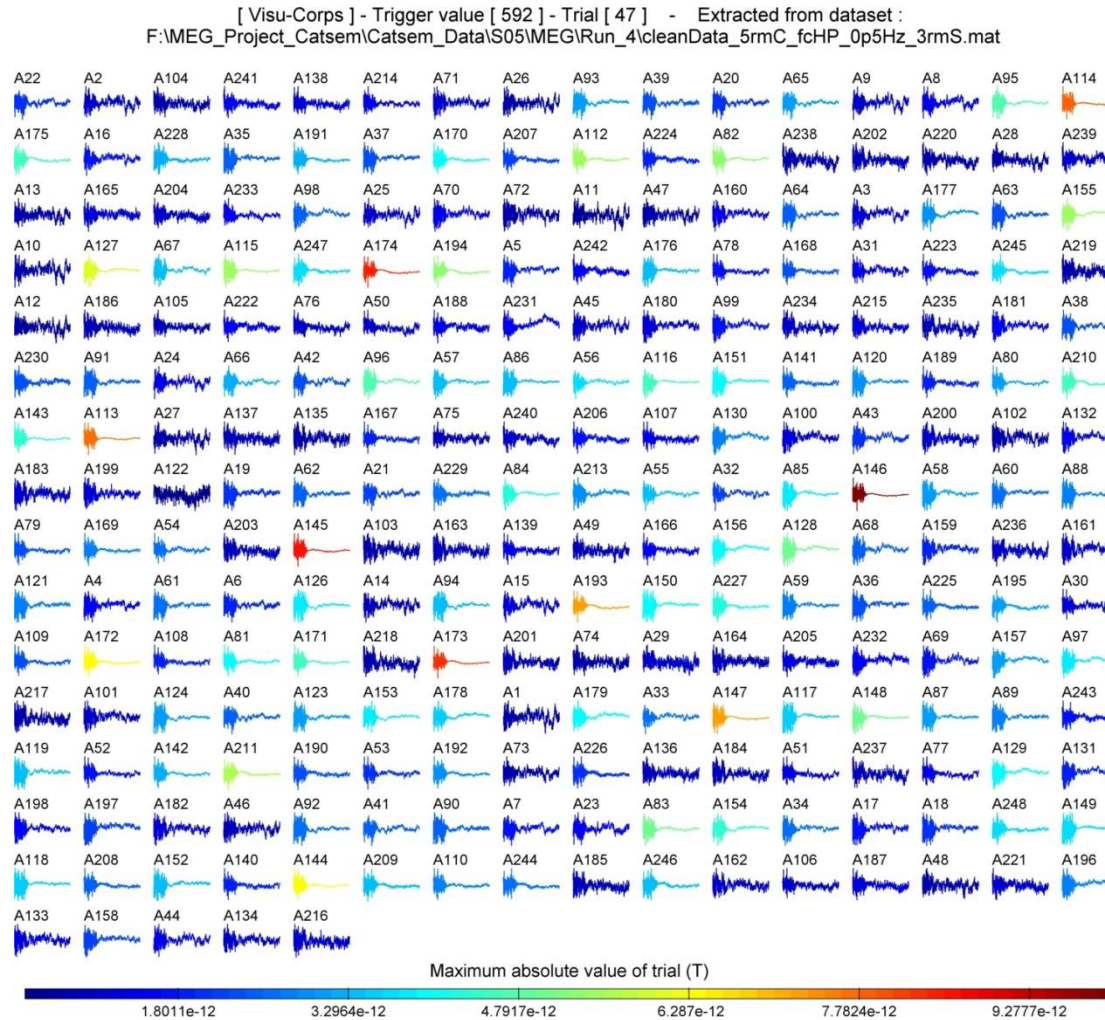
Filtrage

Capteurs défectueux

Artéfacts énergétiques

Nettoyage via l'ICA

Extraction et tri des essais





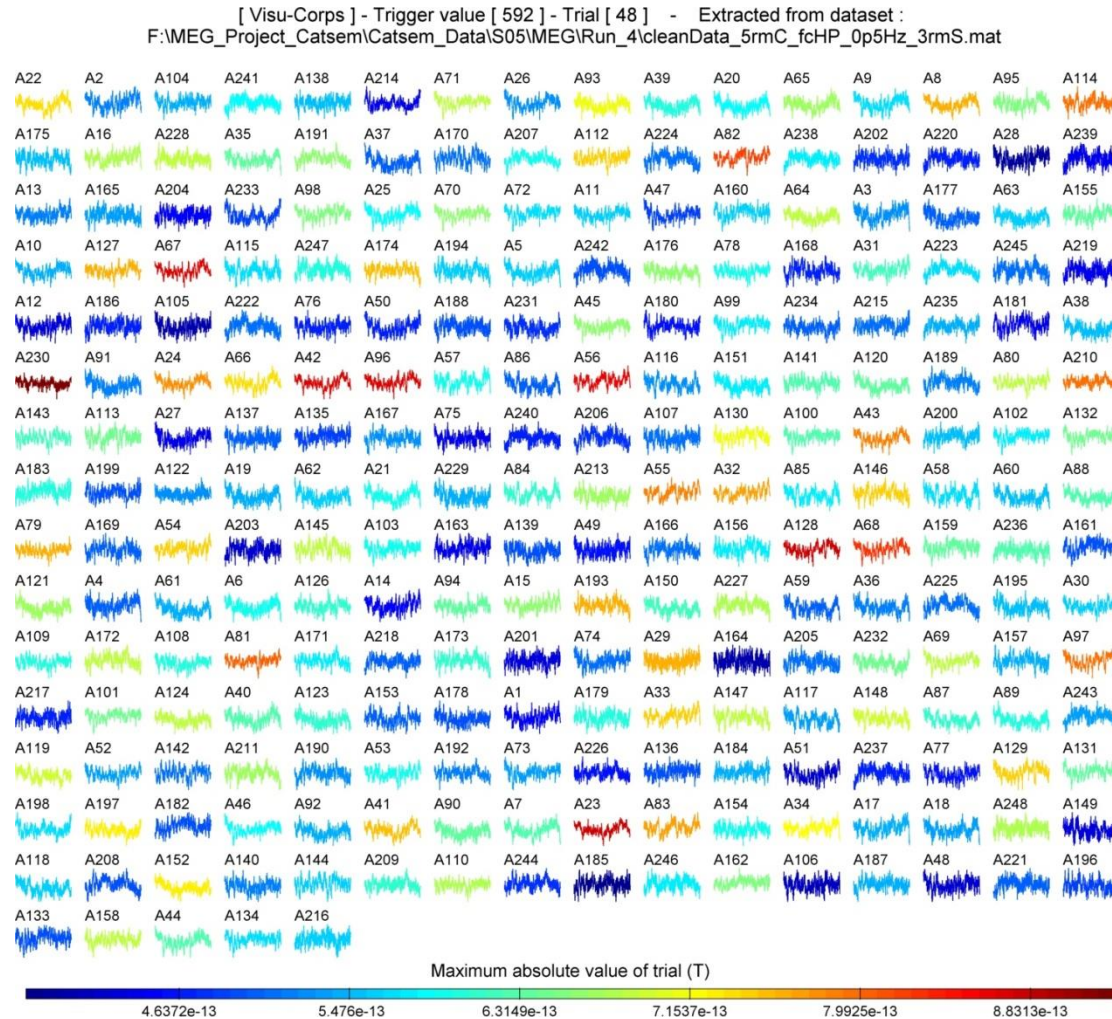
Filtrage

Capteurs défectueux

Artéfacts énergétiques

Nettoyage via l'ICA

Extraction et tri des essais



Filtrage

Capteurs  
défectueux

Artéfacts  
énergétiques

Nettoyage  
via l'ICA

Extraction et  
tri des essais

```
badT = [13 23 40 47]; % Indices des essais à supprimer  
goodT= setxor(1:length(trials.trial),badT); % Bons essais
```

```
cfg = [];  
cfg.trials = goodT;  
cleanTrials = ft_redefinetrial(cfg, trials);
```

Acquisition

Prétraitement

Analyses spatio-temporelles

Champs évoqués et activité induite

Localisation des sources

Connectivité

Statistiques



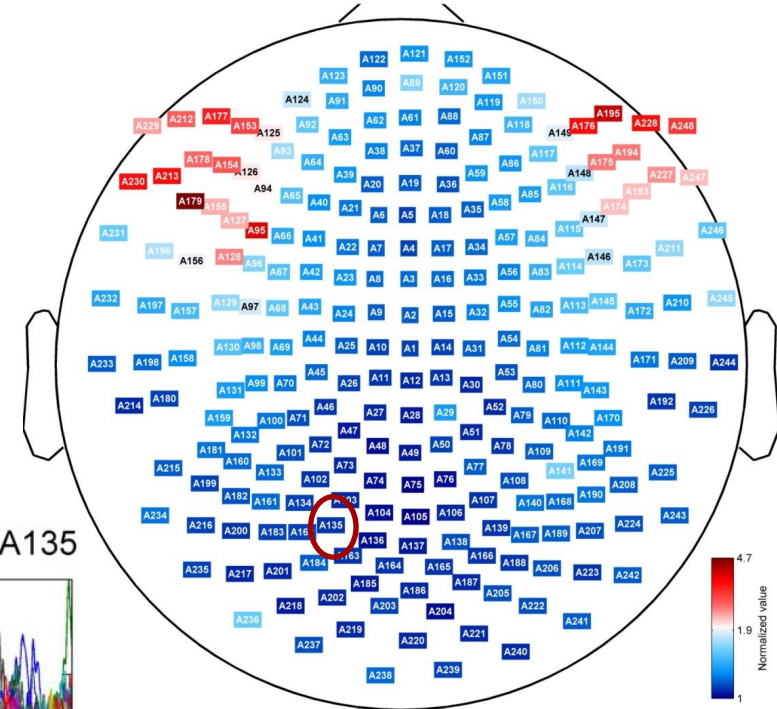
**A partir du jeu de données nettoyé**



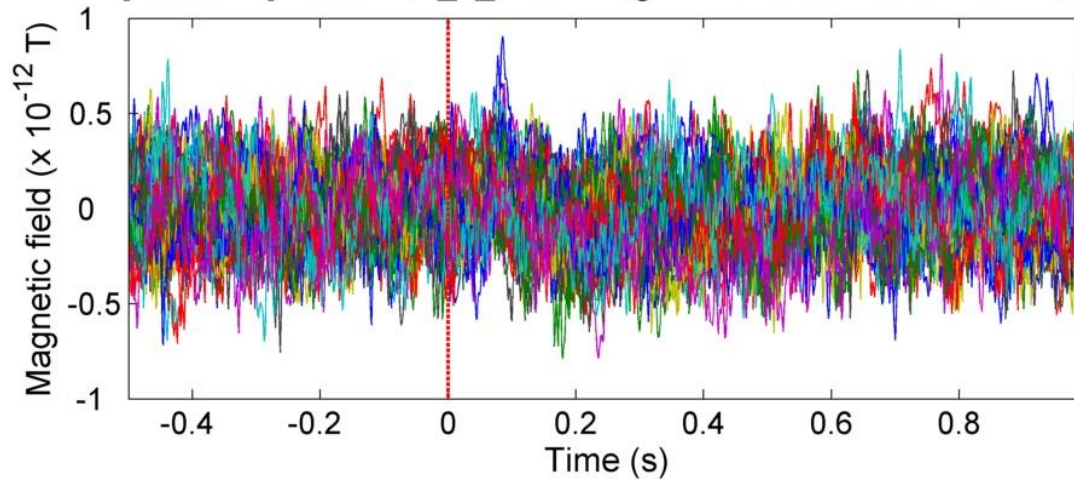
## Moyenne des essais par condition et par capteur

Exemple RegHF S08\Run\_3

F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\ftData\_fcHP\_0p5Hz  
Mean value of signal envelop (from hilbert transform)



[ Catsem ] - S08\Run\_3\_Visu - RegHF - 54 trials - Channel A135

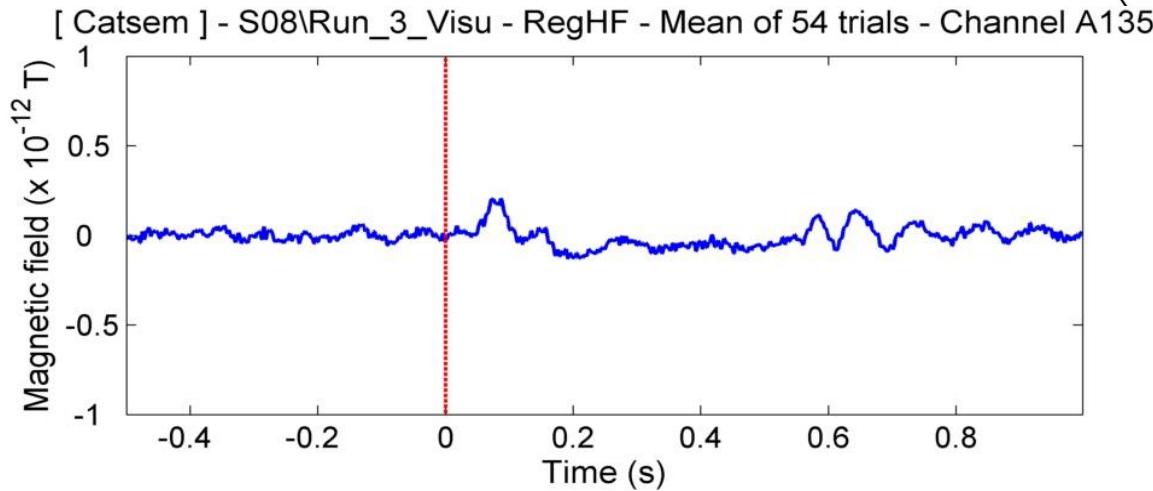
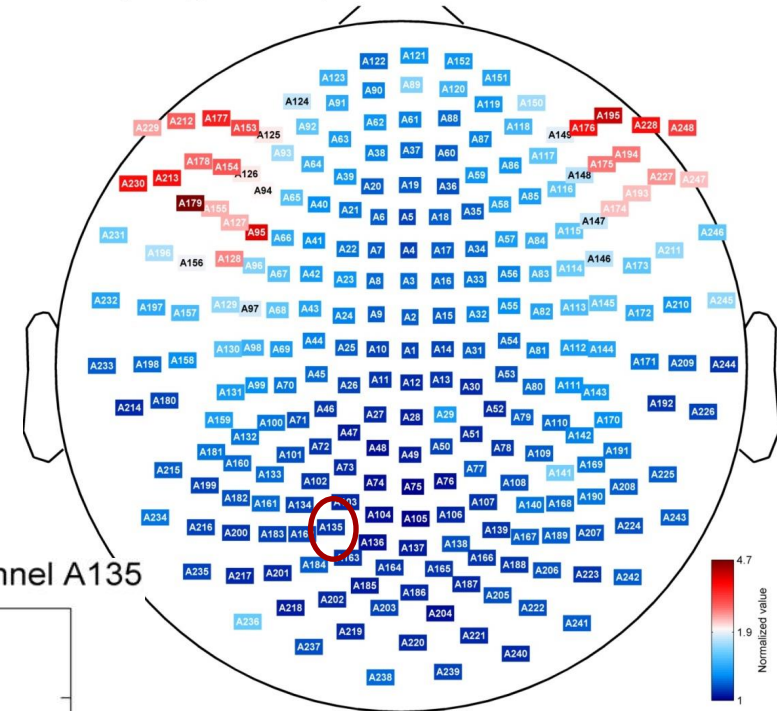


```
cfg = [];
cfg.keeptrials = 'no';
avgTrials =
ft_timelockanalysis(cfg, cleanTrials);
```

## Moyenne des essais par condition et par capteur

Exemple RegHF S08\Run\_3

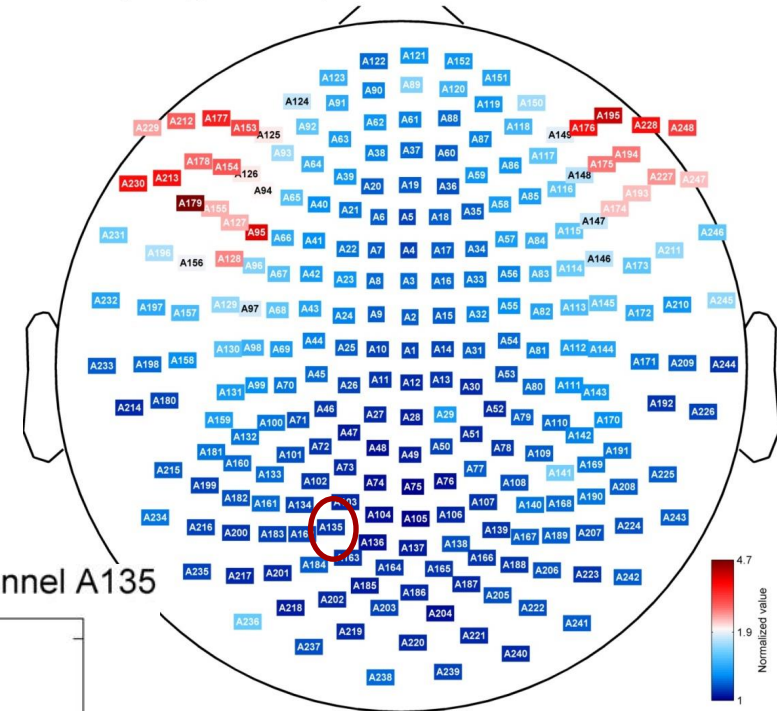
F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\filtData\_fcHP\_0p5Hz  
Mean value of signal envelop (from hilbert transform)



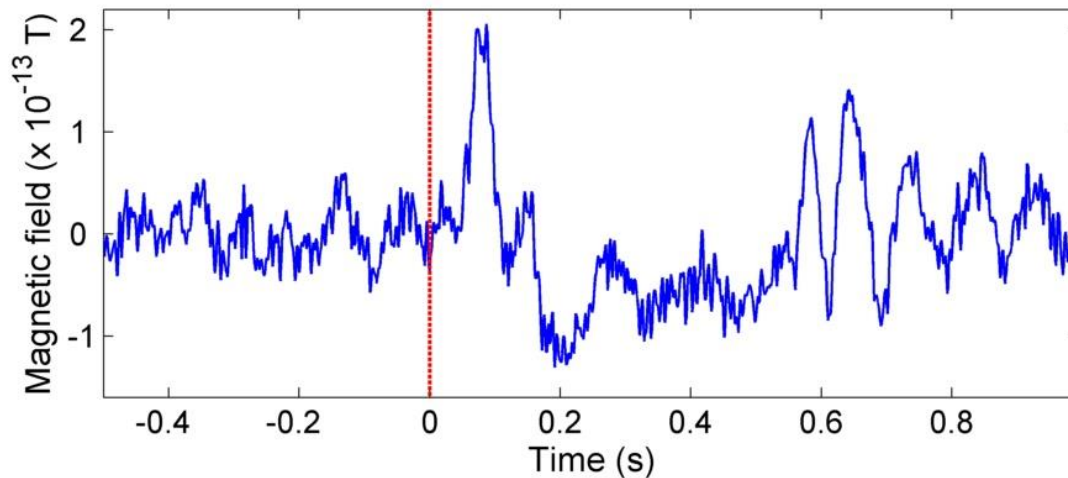
## Moyenne des essais par condition et par capteur

Exemple RegHF S08\Run\_3

F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu\filtData\_fcHP\_0p5Hz  
Mean value of signal envelop (from hilbert transform)



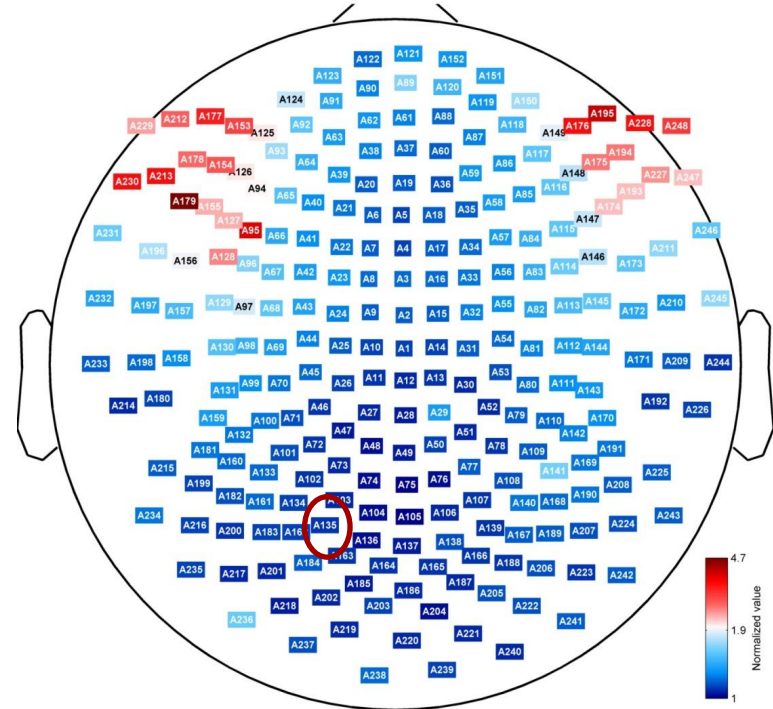
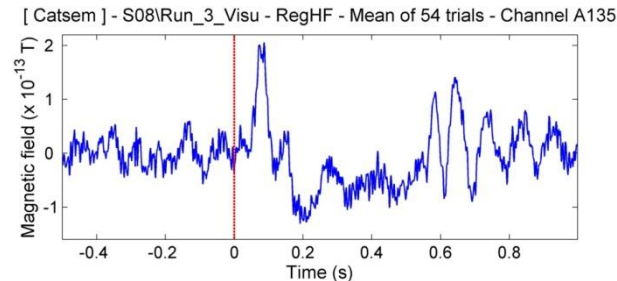
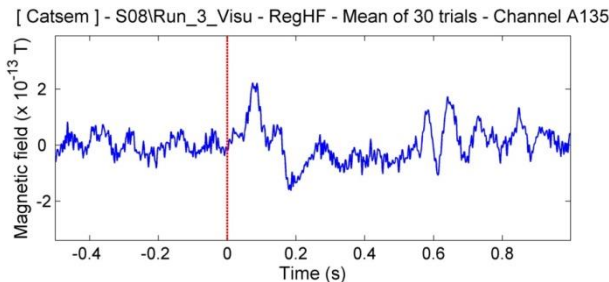
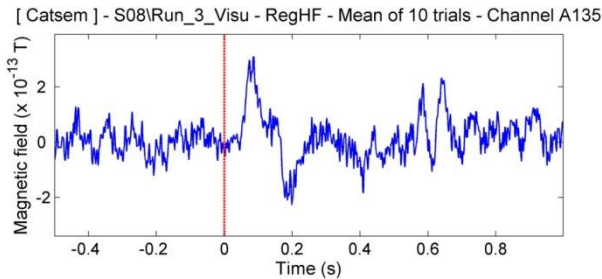
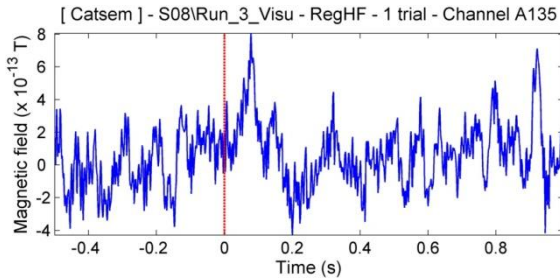
[ Catsem ] - S08\Run\_3\_Visu - RegHF - Mean of 54 trials - Channel A135



## Moyenne des essais par condition et par capteur

### Exemple RegHF S08\Run\_3

F:\MEG\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_VisufiltData\_fcHP\_0p5Hz  
Mean value of signal envelop (from hilbert transform)

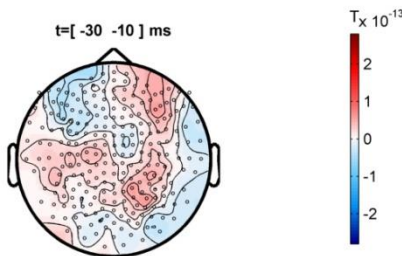


## Représentation topographique à partir des essais moyens par condition et par capteur

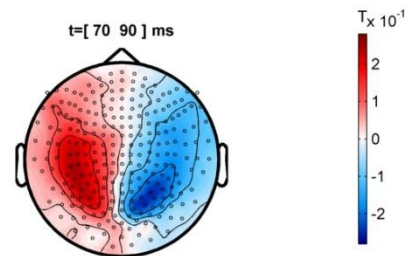
Exemple RegHF S08\Run\_3

`ft_topoplotER`  
`meg_topoER_frame`

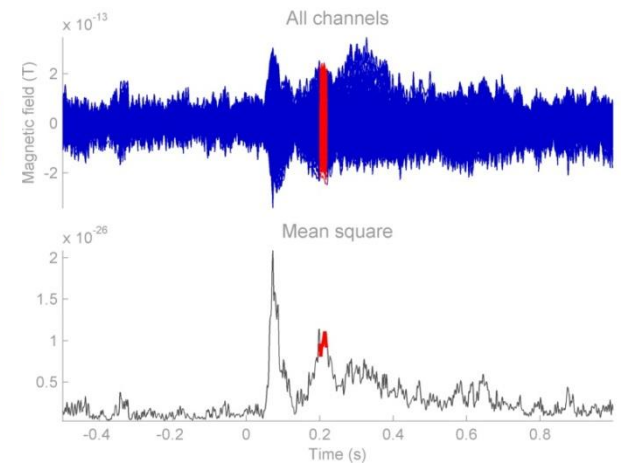
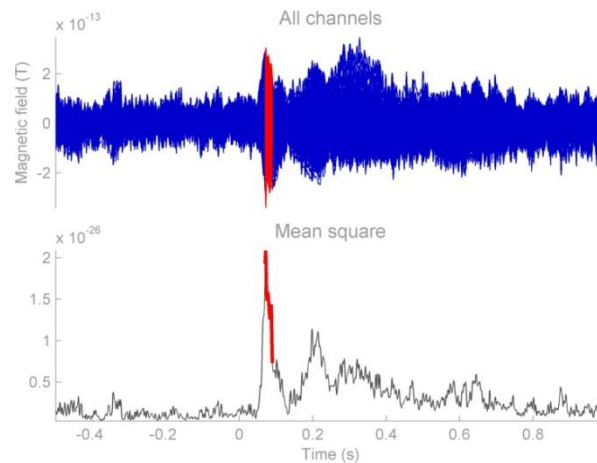
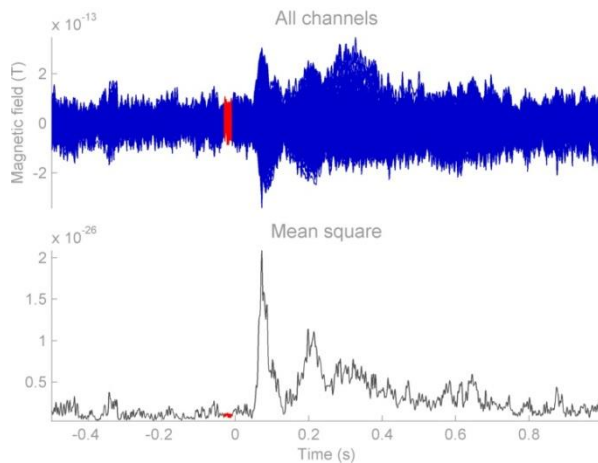
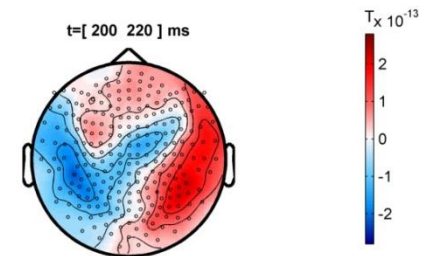
[ Visu\_RegHF ] - Topographic representation of ERF : avgTrials.mat  
F:\MEG\_Project\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu



[ Visu\_RegHF ] - Topographic representation of ERF : avgTrials.mat  
F:\MEG\_Project\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu



[ Visu\_RegHF ] - Topographic representation of ERF : avgTrials.mat  
F:\MEG\_Project\_Catsem\Catsem\_Data\S08\MEG\Run\_3\_Visu





But :

Estimer les sources de courant interne à l'origine du signal enregistré par les capteurs MEG

## ➔ Problème inverse

- Résolution du problème direct :  
Définition d'un modèle de source  
Calcul des champs générés à partir de ce modèle de source
- Ajustement des paramètres du modèle (position, orientation des sources) pour que les champs générés ressemblent le plus à ceux mesurés
- Non unicité de la solution
- Une multitude de modèles & de méthodes de résolution

Les étapes réalisées :

## Problème direct

### Préparation du modèle de tête

- ▶ Réalignement et ajustement au système de coordonnées optimisé pour FieldTrip
- ▶ Segmentation du volume
- ▶ Construction du modèle de tête (volume de conduction)
- ▶ Création de la grille des sources dipolaires basée sur un template (T1.nii)
- ▶ Ajustement de la grille des sources à l'anatomie du sujet
- ▶ Calcul de la matrice de gain (lead field) : la solution du problème direct

## Problème inverse

### Localisation des sources par beamforming

- ▶ Reconstruction des sources par la méthode LCMV
- ▶ Normalisation des sources

## Fonctions utilisées

### Problème direct

#### Préparation du modèle de tête

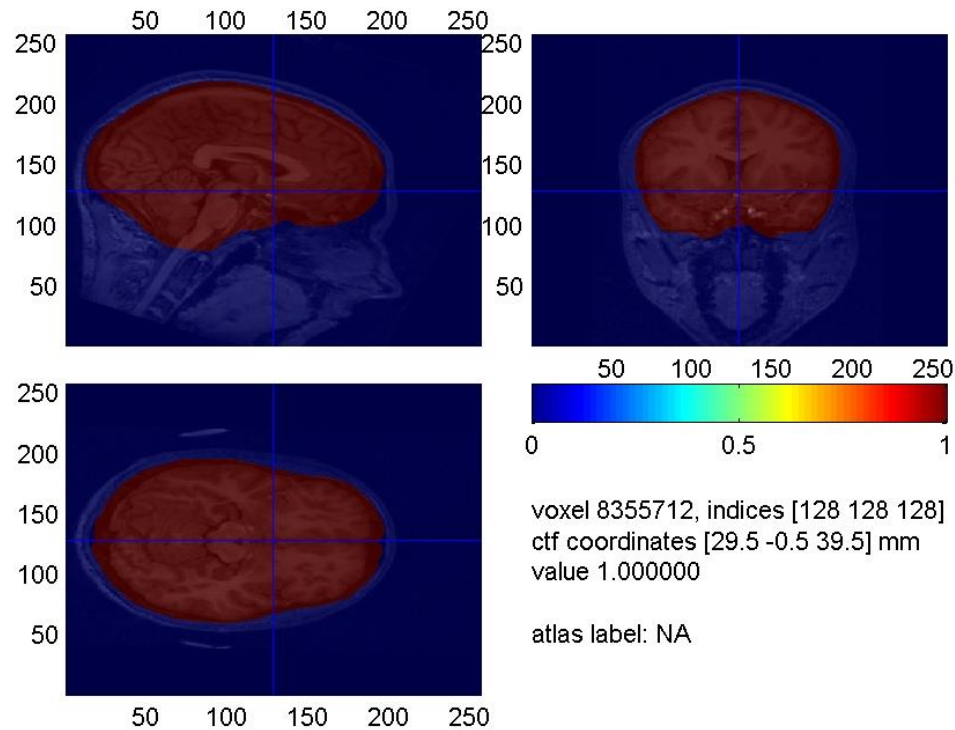
```
ft_read_mri, ft_volumerealign, ft_volumereslice  
ft_volumesegment  
ft_prepare_headmodel  
ft_prepare_sourcemodel  
ft_prepare_leadfield
```

### Problème inverse

#### Localisation des sources par beamforming

```
ft_sourceanalysis
```

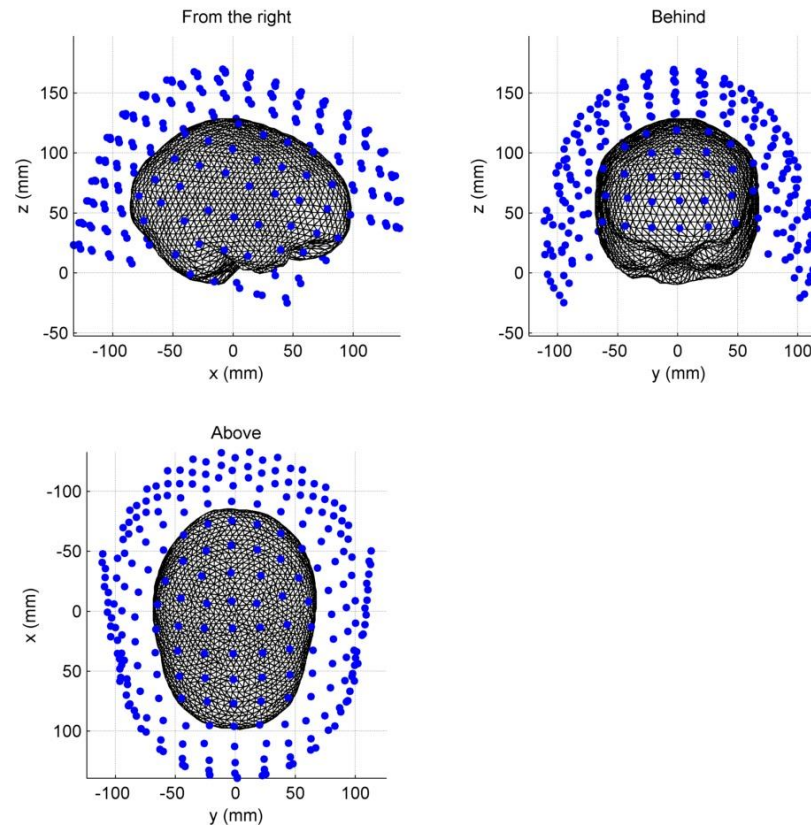
## Problème direct – Segmentation de l'IRM anatomique



```
cfg = [];  
cfg.output = 'brain';  
mriSeg = ft_volumesegment(cfg, mriRes1);
```

## Problème direct – Construction du modèle de tête réaliste

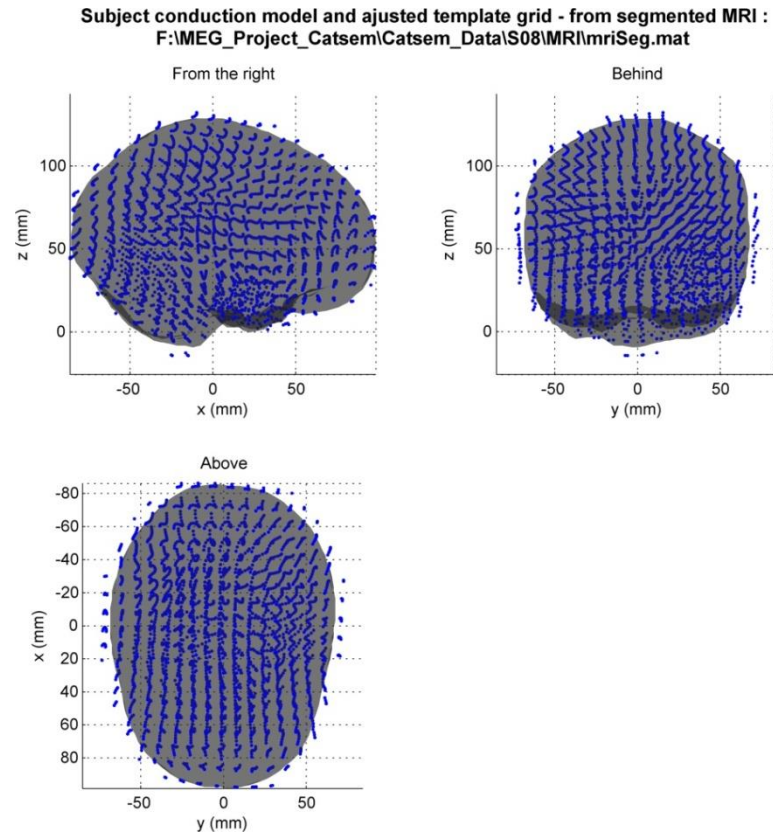
MEG channels & Head model (brain surface) obtained from segmented MRI :  
 F:\MEG\_Project\_Catsem\Catsem\_Data\S08\MRI\mriSeg.mat



```

cfg = [];
cfg.method = 'singleshell';
volmodel = ft_prepare_headmodel(cfg, mriSeg);
  
```

## Problème direct – Déformation de la grille de template



Transformation non linéaire appliquée à la grille du template, qui permet de faire correspondre ses points anatomiques aux points correspondants de l'anatomie du sujet

## Problème direct – Calcul de la matrice de gain (leadfield) : solution

```
cfg = [];  
cfg.grad = cleanTrials.grad;  
cfg.vol = volmodel;  
cfg.grid = subj_grid;  
cfg.reducerank = 2;  
cfg.channel = cleanTrials.label';  
leadfield_grd = ft_prepare_leadfield(cfg);
```

## Problème inverse : reconstruction des sources

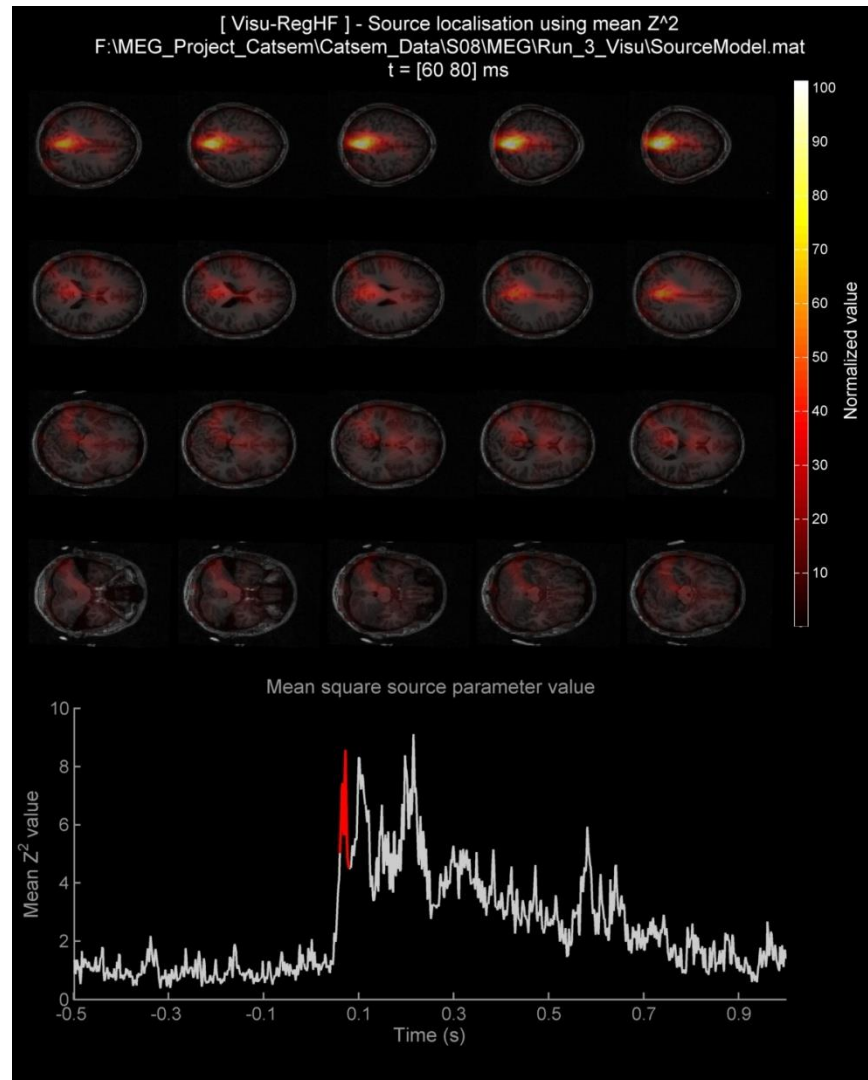


Solution du modèle direct &  
modèle de conduction

```
....  
sourceHF = ft_sourceanalysis(cfg, avgTrials);
```

Estimation d'un filtre spatial à partir des essais moyens (beamforming), utilisation de l'algorithme de minimisation Linearly-Constrained Minimum Variance

## Problème inverse : reconstruction des sources



```
ft_sourceplot  
meg_loc_map_frame
```



Une liste non exhaustive :

- BESA Research
- ASA (Ant Neuro)
- CURRY Scan 7 Neuroimaging Suite
- EMSE Suite (Source Signal Imaging Inc.)
- CTF MEG Software (Instrumentation + Analysis)
- sLORETA/eLORETA (sources localization)
- FreeSurfer / MNE Suite (C, Matlab & MNE-Python)
- SPM8 (Matlab)
- Nutmeg (Neurodynamic Utility - avec SPM8 Matlab)
- Anywave (Laboratoire MEG, INS)
- FieldTrip (Matlab)
- Brainstorm

"FREE"  
↓