

# ProcessMRI : manuel d'utilisation

Florent Grélard

9 octobre 2019

# 1 Présentation

Le logiciel ProcessMRI regroupe des méthodes de traitement d'images IRM. Ce logiciel est particulièrement adapté pour le traitement d'images issues d'acquisitions multi-échos.

Plusieurs méthodes sont implémentées :

1. des méthodes de suppression de bruit Ricien : méthode non-locale et méthode par correction temporelle de phase
2. ajustement de fonctions multi-exponentielles pour le calcul de l'image en densité et en  $T_2$  (ou  $T_2^*$ ).

Développé en Python, il repose sur plusieurs bibliothèques :

- tkinter, pour l'interface graphique
- bruker2nifti, pour la conversion d'images au format Bruker (fichiers `ser` et dossier `pdata`)
- nibabel, pour la lecture et la sauvegarde d'images au format NIFTI.
- scikit, pour l'ajustement de fonctions exponentielles et le débruitage par méthodes non-locales
- numpy, pour la gestion des images et tableaux de données

## 1.1 Aperçu

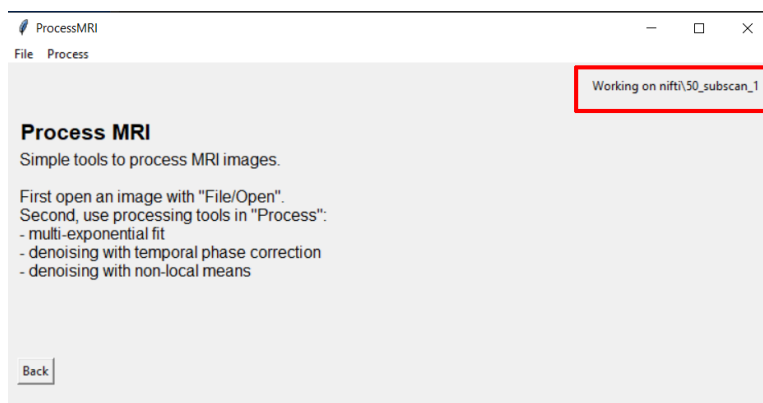


FIGURE 1 – Fenêtre principale et nom de l'image ouverte (encadré en rouge).

La fenêtre principale de ProcessMRI est présentée Figure 1. Les différentes fonctionnalités sont regroupées dans la barre de menu :

- Le menu “File” permet d'ouvrir des images au format Bruker et NIFTI, et de sauvegarder au format NIFTI.
- Le menu “Process” regroupe les méthodes de traitement des images IRM.

Un message dans le coin supérieur droit indique l'**image de travail** sur laquelle les traitements vont être effectués (cf. Fig. 1, encadré en rouge). Ce logiciel n'est pas destiné à la visualisation d'images. On pourra, par exemple, utiliser ImageJ à cette fin.

## 1.2 Lecture et sauvegarde d'images

L'option “File/Open/Bruker directory” permet de convertir les images IRM au format Bruker vers le format image NIFTI. Le dossier à ouvrir doit impérativement contenir un dossier fils nommé `pdata` (cf. Fig. 2).

Après la conversion des images, une fenêtre de dialogue s'ouvre dans un dossier nouvellement créé, contenant les images NIFTI converties. Le dossier est créé à la racine du dossier Bruker ouvert à la

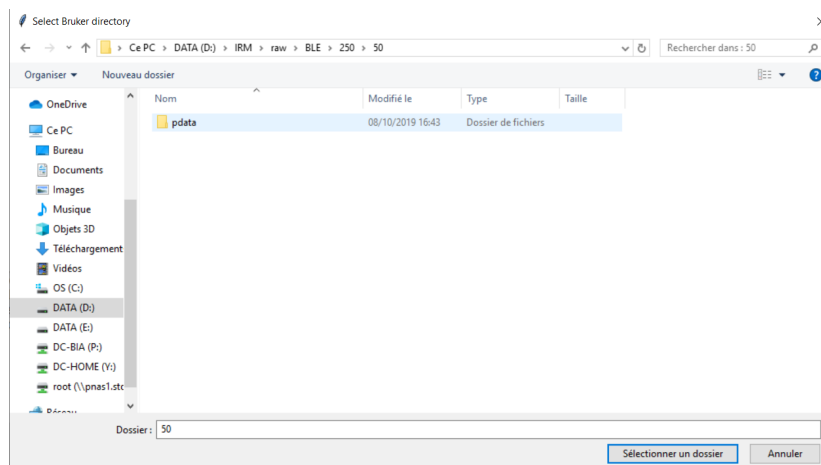


FIGURE 2 – Dossier Bruker pris en charge par le logiciel ProcessMRI : celui-ci contient un dossier fils nommé **pdata**.

première étape. Le nom des images contient le numéro d’étude, ainsi que le numéro de l’acquisition (cf. Fig. 3). Il suffit alors de sélectionner l’image NIFTI voulue (extension “.nii.gz”).

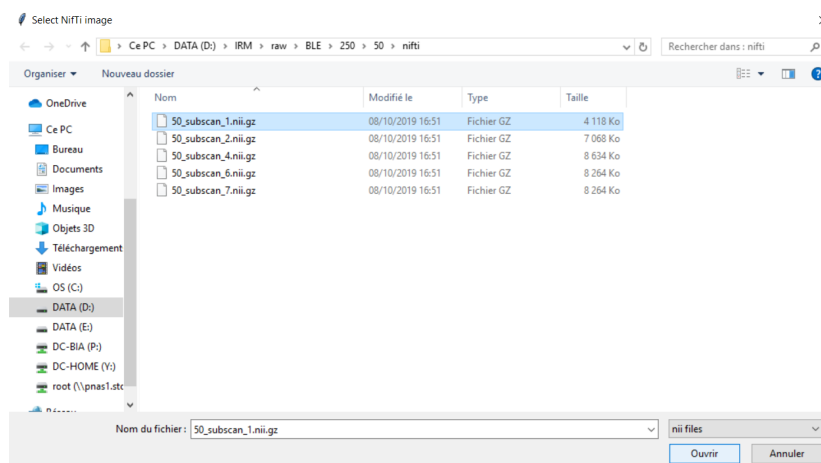


FIGURE 3 – Fenêtre de sélection des images NIFTI converties à partir d’un dossier Bruker. Dans ce cas, le numéro d’étude est “50”, et les numéros d’acquisition “1”, “2”, “4”, “6”, “7”. Il s’agit des sous-dossiers présents dans le dossier **pdata** de Bruker.

Le nom de l’image sélectionnée doit alors apparaître dans l’encart en haut à droite de la fenêtre principale (cf. Fig. 1).

## 2 Traitement

Les méthodes de traitement ont divers paramètres : des informations sur ceux-ci peuvent être obtenues en survolant la boîte ? située à leur droite.

Lors du traitement, une barre de progression apparaît. Un message d’information signale la fin du traitement et indique, le cas échéant, la nouvelle image de travail (cf. Fig. 4).

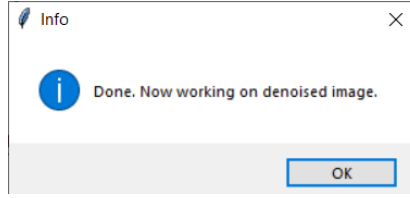


FIGURE 4 – Fenêtre de dialogue lorsqu’un traitement est achevé.

## 2.1 Ajustement de fonctions exponentielles

Cette fonctionnalité permet d’ajuster une fonction multi-exponentielle sur une image IRM multi-échos afin d’estimer une image en densité (à  $t = 0$ ) et en mobilité ( $T_2$  ou  $T_2^*$ ). La fenêtre pour l’ajustement de fonctions multi-exponentielles est présentée dans la Figure 5.

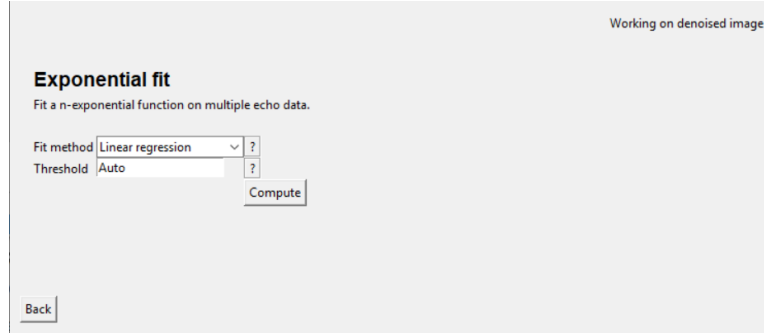


FIGURE 5 – Fenêtre d’ajustement de fonctions exponentielles.

Le paramètre **Fit method** permet de choisir la méthode d’ajustement. **Linear regression** estime les paramètres d’une fonction mono-exponentielle par régression linéaire sur le logarithme des intensités observées. Les valeurs **Mono-**, **Bi-**, **Tri- Exponential** permettent d’estimer les paramètres d’une fonction multi-exponentielle par moindres carrés non-linéaires (NNLS).

Le paramètre **Threshold** permet de fixer un seuil sur les intensités des pixels en deçà duquel l’ajustement n’est pas effectué. Cela permet d’éviter des aberrations d’ajustement pour les pixels de l’arrière-plan. Par défaut, ce paramètre est fixé sur **Auto** : le seuil est estimé automatiquement à partir d’un mélange de gaussiennes sur l’histogramme de l’image. Si l’on ne souhaite pas mettre de seuil, on peut fixer ce paramètre à 0.

Lorsque les paramètres sont choisis, on peut lancer le traitement en cliquant sur le bouton **Compute**. À son issue, une image **density** et **t2\_star** sont enregistrées dans le dossier où se situe l’image de travail. L’image de travail devient alors l’image de densité.

## 2.2 Débruitage par méthode non-locale

Cette méthode de débruitage est basée sur une méthode non-locale. Le principe est d’exploiter la redondance de l’information de l’image. Elles calculent la similarité entre deux points dans l’image en considérant des voisinages, appelés *patches*, autour de ces points. Les auteurs de [1] adaptent une méthode non-locale spécifiquement pour supprimer le bruit Ricien. La fenêtre associée à ce traitement est présentée dans la Figure 6.

Le paramètre **Patch size** correspond à la taille du patch, en pixels. Pour une taille  $n$ , le patch couvre une région de  $n \times n$  pixels. Le paramètre **Patch distance** permet de définir la fenêtre de recherche de similarité des patches dans les images. La similarité est calculée pour des points situés à

FIGURE 6 – Débruitage du bruit Ricien par méthode non-locale.

une distance au point d'intérêt inférieure ou égale à la valeur spécifiée  $d$ , c'est-à-dire que la taille de la fenêtre de recherche est  $2 \times d$ . Le paramètre **Noise spread** correspond à un facteur multiplicatif de la variance du bruit. La variance du bruit est un terme permettant de retrouver les intensités débruitées [1]. Dans l'article, le paramètre est fixé à 1.

Lorsque les paramètres sont choisis, on peut lancer le traitement en cliquant sur le bouton **Compute**. À son issue, une image **denoised** est enregistrée dans le dossier où se situe l'image de travail. L'image de travail devient alors l'image débruitée.

### 2.3 Débruitage par correction temporelle de phase

Cette méthode de débruitage exploite l'information temporelle dans les images IRM multi-échos complexes. L'objectif est de convertir le bruit Ricien en bruit gaussien en corrigeant la phase. La méthode consiste à ajuster un polynôme d'ordre faible sur les valeurs de phase. Les intensités complexes sont multipliées par le conjugué des valeurs de ce polynôme, la partie imaginaire est diminuée : les valeurs de phase approchent 0. La partie réelle contient uniquement le bruit gaussien. Une méthode débruitage classique (filtre médian, méthode non-locale simple) peut alors être employée.

FIGURE 7 – Fenêtre pour la correction temporelle de phase.

Il est nécessaire d'ouvrir une image **complexe** pour utiliser cette méthode.

Le paramètre **Polynomial order** permet de spécifier l'ordre du polynôme à ajuster. Dans l'article, ce paramètre est fixé à 4, pour 48 échos. Ce paramètre est à ajuster en fonction du nombre d'échos dans l'image. Le paramètre **Noise threshold** permet de spécifier un seuil pour ne considérer que la région d'intérêt dans l'image, et ne pas effectuer l'ajustement sur des pixels de l'arrière-plan.

Lorsque les paramètres sont choisis, on peut lancer le traitement en cliquant sur le bouton **Compute**. À son issue, quatre images **real\_tpc** (partie réelle), **imaginary\_tpc** (partie imaginaire), **magnitude\_tpc**

et `phase_tpc` sont générées dans le dossier de l'image de travail. L'image de travail devient alors l'image réelle.

## Références

- [1] Nicolas Wiest-Daesslé, Sylvain Prima, Pierrick Coupé, Sean Patrick Morrissey, and Christian Barrillot. Rician Noise Removal by Non-Local Means Filtering for Low Signal-to-Noise Ratio MRI : Applications to DT-MRI. In Dimitris Metaxas, Leon Axel, Gabor Fichtinger, and Gábor Székely, editors, *Medical Image Computing and Computer-Assisted Intervention MICCAI 2008*, volume 5242, pages 171–179. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.