

Tractography Classification

Table of Contents

Introduction.....	1
Create ROI.....	1
Prepare the anat.....	1
Coregistration anat to DWI.....	1
Normalization & Segmentation of the anatomical volume.....	2
Prepare the ROI mask.....	3
Subcortical.....	3
Cortical roi from MNI.....	3
Subcortical then end.....	6
be careful with your header.....	7
Seed to target Classification.....	7
Connectivity-based segmentation.....	8
Connectivity-based segmentation in the anat space.....	10
Blind classification : seed to white matter.....	10
Seed to target Classification with mrtrix.....	12

Introduction

We are working on DATA_TP/Sujet1. You have the data of one subject with an anatomical scan (T1 mp2rage 1mm iso) in the anat subdirectory and a diffusion acquisition in the DWI subdirectory (a 2mm iso EPI acquisition with 60 direction at b=1500 apply in 2 opposition phase encoding direction AP PA).

The subdirectory with a prefix _done, contains all the results of the different process you will do during this practical.

Create ROI

Prepare the anat

Since we will define roi from the anatomical scan and use them for Diffusion tractography, it is important to correct for motion between the 2 sessions (anat and DWI). We will then normalize the anat in order to be able to transform roi from MNI standard space to the subject space.

Coregistration anat to DWI

Before to run the coregistration go to the anat directory in a terminal and type

```
fslhd s*nii
```

A fsl command that display the nifti header. Nifti header contains 2 matrix of orientation : the qform and the sform. It has a name and a matrix (4x4) that code for the volume position in millimeter space. (it related the i,j,k voxel of the 3D data box grid to a x,y,z millimeter coordinate in space).

Note the qform_name and sform_name and note that the 2 matrix are identical. A similar information can be obtain with mrtrix command

```
mrinfo s*nii
```

the information is summarized and only one matrix is display. This is the sform matrix that is used by default. (this is true for most of the software fsl freesurfer mrtrix ... but this is not always the case, and it can be confusing)

Run SPM

From SPM main windows :

```
choose coregister -> estimate (first of the list)
```

set the inputs :

```
Reference Image : choose DWI mean B0 ( ../DWI/B0_mean_4D_dwieddycor.nii)
```

```
Soruce Image : choose the anatomical volume
```

Once it is done go to the next paragraph to run the segmentation (and come back here while the segmentation is running)

Compare now the nifti header after spm coregister.

```
fslhd s*nii
```

This is an important point and a major difference to other similar tool (like flirt from fsl) : spm apply the displacement directly to the header of the volume. Note that the data has not been resampled : just the header has been changed. The orientation has been adjust in order to compensate the subject movement between the 2 sessions.

to undo the registration use fsorient utils : (from the anat dir)

```
cp s_S02_t1mpr_SAG_Nsel_S176.nii anat_NOCOREG.nii
```

```
fsorient -copyqform2sform anat_NOCOREG.nii
```

```
fsorient
```

like all other command typing the command without argument will display the help and you can see and understand all the options. Note that for mrtrix command line it is display like the bash more function. Use the “arrow” or the “space bare” to display the next section, and “q” to quit.

```
mrinfo
```

```
fsmaths
```

```
fslmaths | more #to
```

Redo the same with “coregister and reslice” from the anat_NOCOREG.nii (which is the same image with same header than the original s_S02_t1mpr_SAG_Nsel_S176.nii).

Understand the difference.

→ the nifti header of anat_NOCOREG.nii is modified (by estimate) and a new image is resliced in the space of the target image.

```
mrinfo s*nii ranat*nii
```

it creates a volume ranat*.nii (the resliced version which has now the same dimension and matrix orientation as the DWI volume) check that the s*nii has a new sform like previously, it is coregister with DWI. Note that if you do twice the coregister, spm will copy the sform to the qform before changing the sform to apply the new coregistration. You will loose the original orientation. (this is not a big deal since only the sform it taken into account, but be aware)

Visualise the results:

with spm :

choose "Check Reg" and select the volumes you want to compare
with fslview (from the anat dir)

```
fslview s*nii ../DWI/ B0_mean_4D_dwieddycor.nii
```

this does not work because fslview only load volumes with the same matrix size

```
fslview ranat*.nii ../DWI/ B0_mean_4D_dwieddycor.nii
```

Change the colormap of the B0 volume. switch between volume with selecting and unselecting the "eyes icon" near the first volume name. You can also change the transparency level. this works but not very nice to check since we are in DWI space. Try now with mrview which has the advantage to properly handle different matrix size (and it takes into account the nifti header !)

```
mrview s*nii ../DWI/B0_mean_4D_dwieddycor.nii
```

type "F2" in the mrview windows

choose Tool-> overlay and open the B0_mean_4D in orverlay

activate the Threshold box and choose 100 in the first value (only value above 100 will be display

add the volume anat_NOCOREG.nii

use arrow "pageUP" to switch between volume. Go near the ventricle to see the improvement of coregistered volume.

Compare with fslview

```
fslview s_*nii anat_NO*nii
```

No difference because fslview doesn't take into account the nifti header !

You do not see a displacement (unlike with mrview). This is an important fact of fslview : fslview view just open the volume box without taking into account the header.

Normalization & Segmentation of the anatomical volume

For SPM main windows :

choose Segment

choose the input

Volumes : chose the s_*nii

change some defaults to get also normalized tissue (c1 / c2 / c3 : gray / white /csf)

change for tissue 1 2 and 3 : Warped Tissue : Modulated %to directly get the normalized c1 c2 c3

Deformation Fields : choose "Inverse + Forward"

The last point is important to be able to apply the normalization to other volume, or to de-normalise volume from the MNI space into the subject space. This will produce 2 deformations fields y_s*nii (the direct one subject to MNI) and iy_s*.nii the inverse.

You can go one while it is computing (10 to 15 mn) or come back to the previous paragraph.

Prepare the ROI mask

Subcortical

Fsl subcortical segmentation. <http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FIRST/UserGuide>

Copy the T1 volume (s_*.nii) to a sub-diretory and run the automatic segmentation with first.

Instead to do a copy (: cp ../s*nii), we make a symbolic link to save disk space

```
cd Sujet/anat
mkdir first
cd first
cp ../s_*nii .
run_first_all -i s_*nii -o first_
```

You can go one while it is computing (10 to 15 mn)

Cortical roi from MNI

Look at the aal atlas in a terminal

```
cd DATA_TP/ROI_mni/all
gedit aal.nii.txt
fslview aal.nii
```

We want to regroup some label of the atlas into 6 different ROI. We will use a octave script that will make and run the c3d command. C3d is a command line tool that perform basic computation on nifti volume, similar to fslmat (from FSL) or mrcal and mrmath (from mrtrix), or image_calc (from spm). <https://sourceforge.net/p/c3d/git/ci/master/tree/doc/c3d.md>

Run octave in a terminal

```
cd DATA_TP
octave
```

add to your path the script directory

```
pwd
```

```
addpath /DATA_VM/DATA_TP_roman/script
```

```
roiname =
{'aal_precentral', 'aal_postcentral', 'aal_parietal', 'aal_ssociatif', 'aal_temporal',
' , 'aal_occipital'}
label = {2, 58, [60 62 64 66], [4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36],
[82 84 86 88 90], [44 46 48 50 52 54 56]}
dir_roi = {pwd}
v_brodmaan={fullfile(dir_roi{1}, 'aal.nii')}
write_multiple_mask(v_brodmaan, label, roiname, dir_roi, '3dcalc')
```

Change “copy_paste” by the full path of DATA_TP directory. Note the c3d command that has been done. With a lot of label to concatenate it is useful to use the octave script.

Now we have our 6 roi (for the right hemisphere) we need to de-normalise them to the subject space using spm normalization (result of the segment function). If it is still running you can open a new spm and use the iy_s*nii from the anat_done subdirectory.

First we copy the mni roi in the subject subdirectory, because spm will write denormalise roi in the same directory and you may have multiple subject ...

```
cd DATA_TP/Sujet1
mkdir roi
cd roi
```

#just to check we select the correct file

```
ls ../../ROI_mni/aal/aal_*nii
```

use the arrow to recall the last command (“CRTR+a” to go at the beginning of the line) and change ls by ln -s to make symbolic link. Add “.” at the end, this mean current directory

```
cp ../../ROI_mni/aal/aal_*nii .
```

Now let's denormalize the ROI: I SPM main windows select

```
Normalize->write
```

```
Data → specify
Deformation field : chose the iy*.nii in the anat directly
Images to Write : choose the 6 aal_*.nii volumes in the roi directly
Bounding box : specify [NaN NaN NaN;NaN NaN NaN]
Voxel size : specify NaN NaN NaN
filename Prefix : specify iw
```

The NaN tells spm to take the destination box and voxel size. Since we are using the inverse deformation field this means the native T1 volume.

Look at the results in fslview or mrview the T1 volume with the roi overlay

Since there are quite big and contain a lot of white matter, we want to clean up and have only the gray matter part. Easy, just multiply each roi (iwaal*.nii) with the segmented gray matter (the c1*.nii volume in the anat dir)

type mrcalc to see the doc. the syntax we want to use is :
mrcal vol1 vol2 -multiply output.nii

let's do it with a for loop to spare some painful copy paste. Note the special quote ` (AltGr+7)

In a terminal in the roi dir

```
for k in `ls iw*`
do
echo $k
done
```

check you select the correct volumes

recall the last command (with the up arrow). and change

```
echo $k
```

with

```
mrcalc $k ../anat/c1s_S02_t1mpr_SAG_Nsel_S176.nii -multiply c1_$k
```

Do not copy past the path but use the tabulation to complete the c1 volume name

```
../a+TAB"/c1+TAB".
```

```
for i in `ls iw*`; do echo mrcalc $i ../anat/c1s_S02_t1mpr_SAG_Nsel_S176.nii
-multiply c1_$i ; done
```

so the output name will have a c1_ prefix

Recall the last command and remove the echo, so that the command is now run

```
for i in `ls iw*`; do mrcalc $i ../anat/c1s_S02_t1mpr_SAG_Nsel_S176.nii
-multiply c1_$i ; done
ls -ltr
```

Redo the same but using fslmath : the syntax is slightly different :

```
fslmaths vol1 -mult vol2 output
```

we choose a prefix fsl_c1_ to see the difference

```
for i in `ls iw*`; do fslmaths $i -mul ../anat/c1s_S02_t1mpr_SAG_Nsel_S176.nii
fsl_c1_$i ; done
ls -ltr
```

The results are the same But as you can see fslmath complains !

Note that the default of fsl is to write nifti gz file format. This is a good practice. Type

```
du -sh ./*
```

roi are big volume full of zero. So the .gz file is only a few kb whereas the .nii is 12M so it takes 1000 times more space on your computer. With a lot of subjects and a lot of roi, you can spare a few

hard disk just typing the following command

```
gzip iwaal*nii
```

the drawback is that you need to unzip them to handle them with spm, but it worth the effort.

(gunzip *gz to unzip all)

Let's go back to the warning and follow the advise of fslmaths : check the orientation of iw*nii volumes, and compare to the original anatomical volume (or the derived volume c1_*nii). you can notice a small bug in the spm inverse normalization. The matrix information is different (ie they have a different voxel grid)

Actually the iw*nii volumes are correct (ie they are coherent with the anat initial volume) but the voxel grid is different (ie a different orientation matrix). So the solution is to reslice the iw*nii roi into the same box as the s_*nii.

The c1_iw* and fsl_c1_iw* are identical but false. Because when you multiply to volume (with fslmath or marcal) this is done in a voxel by voxel basis (ie in the matrix space without taking into account the orientation). So if the volume are coregistered but in a different grid the result is false.

So let's remove them. Do not try to visualize them the difference is too small. If you want to understand the difference go to the next next paragraph (“careful with your header”) where we amplify the difference.

```
rm -f c1_iw* fsl_c1_iw*
```

Let's now reslice the iw*nii roi in the anat space (s_*nii)

For this you can use spm coregister-> write function or directly in command line with flirt tools from fsl.

```
flirt -in iwaal_precentral.nii -ref ../anat/s_S02_t1mpr_SAG_NSel_S176.nii -o
ranat_iwaal_precentral.nii -usesqform -applyxfm -interp nearestneighbour
type flirt to understand the option with set (-in -ref -o -usesqform -applyxfm -interp
nearestneighbour)
```

Use a for loop to do it on all the volume (flirt : commande fsl pour recalage linéaire)

```
for i in `ls iw*`; do echo flirt -in $i -ref
../anat/s_S02_t1mpr_SAG_NSel_S176.nii -o ranat_$i -usesqform -applyxfm -interp
nearestneighbour; done
```

remove the echo to perform the command and check the last results with

```
ls -ltr
```

Now do again the fslmaths command but change the input : instead of ls iw* select the reslices roi : `ls ranat_iwaal*` and just add a c1 prefix for the output

```
for i in `ls ranat_iw*`; do fslmaths $i -mul
../anat/c1s_S02_t1mpr_SAG_NSel_S176.nii c1_$i ; done
ls -ltr
```

Now we have the ROI defined in the subject space (coregister with DWI) and define only for the gray matter. For fsl tracking it can be useful to have those roi in the DWI grid. Since the coregistration has already been made we just need to do a reslice :
we use the same flirt command but we change the input (c1_ranat_iw*), the reference is now the B0 mean volume and the prefix for output (“rdti_”)

```
for i in `ls c1_ranat_iw*`; do flirt -in $i -ref
../DWI/B0_mean_4D_dwieddycor.nii -o rdti_$i -usesqform -applyxfm -interp
nearestneighbour; done
ls -ltr
```

Visualize the results in mrview loading the anat and the mean B0 and select one roi in anat space and on in dti space

Subcortical then end

The first command should now be ready. Find the terminal or go to the first dir (anat/first) the end result of interest for roi definition is :

```
first_2015_07_09_PAS_AVC_Temoin001_all_fast_firstseg.nii.gz
```

this is a label volume

```
fslview ../s_S02_t1mpr_SAG_Nsel_S176.nii
first_2015_07_09_PAS_AVC_Temoin001_all_fast_firstseg.nii.gz
```

and change the color map of the first_ volume, locate the righth thalamus (on the left) and note its value (49). We want to extract a roi as previously for the aal atlas. Since we have only one label we can do it quickly directly with fslmat (or mrcacl)

```
fslmaths first_2015_07_09_PAS_AVC_Temoin001_all_fast_firstseg.nii.gz -thr 48.5
-uthr 49.5 -bin rthalamus
```

this select voxel value greater than 48.5 (-thr) and smaller than 49.5 (uthr) the -bin option set the value of the output to one.

We will also need this roi in DTI space. Easy now with the flirt command

```
flirt -in rthalamus.nii.gz -ref ../../DWI/B0_mean_4D_dwieddycor.nii -usesqform
-applyxfm -o rdti_rthalamus.nii -omat T1_to_DTI.mat
```

We will also need the transformation matrix from T1 to Diffusion, which is obtained by the same flirt command but adding the -omat option (to output the matrix transformation)

be careful with your header

This is optional (if you have time).

Just to come back at the problem we have seen with the inverse normalization of MNI roi into subject space. We notice a different header than the original anat volume.

To make the difference very visible we rotated the anatomical scan (see the anat_rot subdirectory) and we performed the previous analysis again. We omit of course the coregistration to the B0 which will remove our manual rotation. All the roi denormalization are in the roi_rot subdirectory

Look at the data

```
fslview anat_rot/s_S02_t1mpr_SAG_Nsel_S176.nii
anat_done/s_S02_t1mpr_SAG_Nsel_S176.nii
```

the 2 volumes are identical, because we only modify the header which is not taken into account in fslview

```
mrview anat_rot/s_S02_t1mpr_SAG_Nsel_S176.nii  
anat_done/s_S02_t1mpr_SAG_Nsel_S176.nii
```

the volume are in different location but you do not see the tilt

```
open now with spm-> corregister.
```

Now you see the tilt. This is because spm referential is the millimeter space.

mrview convention is between both they choose a millimeter space but they show the plane without the rotation (they are physicists and they want to know the acquisition plane and to avoid any interpolation). So when you switch between volumes of different location it take the slice that goes through the focus point. So only the focus point is at the same position in space (because both volume has a different plane selection). This is why to check a coregistration it is better to use the overlay (the overlaid volume is reslice to match the background volume)

in mrtrix with the anat_rot/ load in the roi editor tool, the 2 different roi :

```
roi_rot/iwaal_precentral.nii.gz and roi_rot/r_anat_iwaal_precentral.nii.gz
```

they look coherent because mrtrix is taking into account the header information. If you do tracking with mrtrix you could choose any of those 2 rois. Mrtrix (like spm) is taking the nifti orientation (except for basic image operation (multiply))

```
load the 2 roi in fslview.
```

They are different. The iwaal is bad because it has a different matrix orientation (so a different grid). The r_anat_iwall is correct because we reslice it into anat_rot space.

If you do tracking with fsl you should use the r_anat one.

Now if you do a multiplication (or roi value extraction) you need to use volume from the same grid if not, bad result may happen without you notice them !!!

load now the fsl_c2_iwaal (which come from the product of iwaal and c1 image) to clearly understand how bad the results can be !

Seed to target Classification

We will now work on the DTI acquisition. Go in the DWI directory and look at the data. The 4D_dwi.nii.gz is the raw acquisition 60 direction acquired twice with opposite phase direction AP PA. In the subdirectory topup the 4D_B0.nii.gz contains the b=0 acquisition, and the unwarped_4D_B0_topup.nii.gz volume is corrected for distortion with topup. We then run eddy to obtain the corrected 4D_dwi_eddy.nii.gz volume. We then run bedpostx in f507_bedptx.bedpostX subdirectory. We are then ready to do some tracking with protract2 fsl command.

Connectivity-based segmentation

The objective of this type of analysis is to ask the following question: "For each location in the seed mask (thalamus), what is the probability of connection to each of my target masks". The output is therefore a single image for each target mask in which only voxels within the seed mask contain data. (voxel values are the number of fiber that reach the specific region)

We first perform this classification in the DTI space so we will need the thalamus and the 6 cortical regions in the dti space (the one we made with prefix "rdti_")

In a terminal open fsl

```
mkdir probtrack
fsl
choose "FDT diffusion"
BEDPOSTX directory : DWI/f507_bedptx.bedpostX
Seed Space single mask : choose the rdti_rthalamus.nii.gz in the anat/first
direction
Optional Targets : select Classification targets. And add the 6 corticale roi
(rdti_c1_ranat_iwaal*nii in the roi subdirectory
Output directory : chose the DWI/probtrack and add at the end a name (for
instance classif_rthalamus)
In advanced Options : check "Use Distance correction"
choose Go : to run
```

Go now in the output directory and look at the probtracks.log

```
cat probtrackx.log # or gedit probtrackx.log
```

you see the command line that could be run directly from the terminal. Type

```
probtrackx2
```

to see the help on the command and understand the options of the previous command

once it is done go in the probtrackx output dir run :

```
find_the_biggest seeds_to_c1_* biggest
```

this output a label image biggest.nii.gz with value between 1 and 6, which correspond to the alphabetical order of the seeds_to* files. So label 1 is the region where the maximum number of fiber reach the associative cortical roi (maximum compared the number of fiber that reach other cortical roi from the same voxel)

While it is running, go to the DWI/probtrac_done/classif_thalamus directory where the analysis has been done. Look at the results in fslview.

```
fslview seed_to* biggest fdt_path ../../roi/rdti*
```

change the color map to understand each seed_to classification, you can also change the Min value to do not display poorly connected voxels. play with the overlay or visibility. You can also load a target mask and the fdt_path. The fdt_path contain the number of fiber the cross each voxel (in the allbrain mask of analyse). This correspond to a probability of the diffusion path starting from the seed.

There is also a file called waytotal which contains the total nuber of fiber that were selected. Fsl is doing (by default) 5000 tirages. Since we put include condition, some fiber may not be selected.

```
cd Sujet1/DWI/probtrac_done/classif_thalamus
```

```
cat waytotal
```

```
3945000
```

compare to the number of voxel of our seed

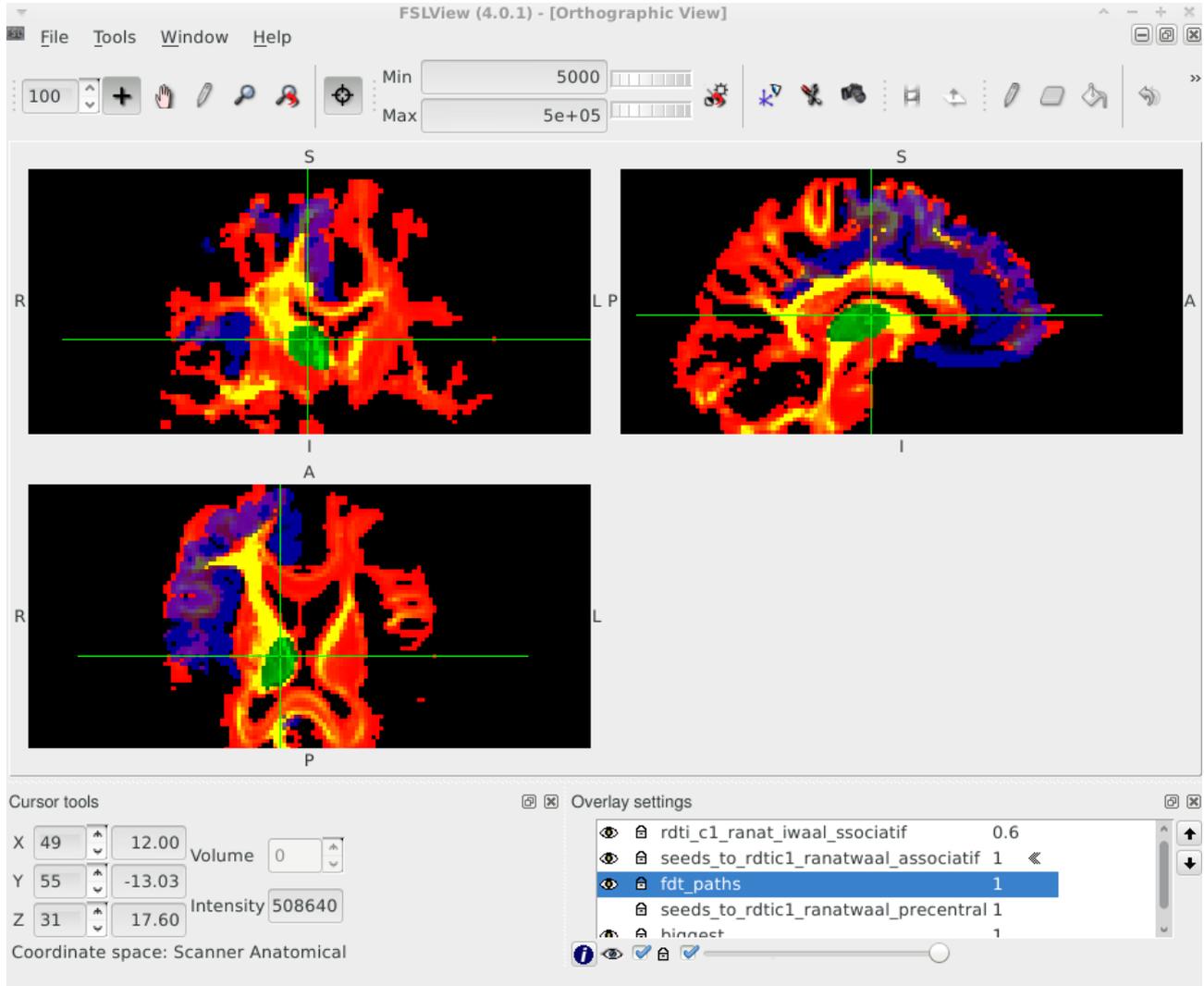
```
fslstats ../../anat_done/first/rdti_rthalamus.nii.gz -V
```

```
1858 14864.000000
```

So $3945000/(1858*5000) = 0.42$. So only 42% of fiber that start from the thalamus reach our define target. Did we forget some important cortical region, or is it just because a lot of fiber just stop

before reaching the cortex ? To know you must run a probtrackx2 command without inclusion criteria. And compare the fdt_path to see the difference.

This is done in Sujet1/DWI/probtrac_done/seed_thalamus. There is no difference (same waytotale) so the 42% comes from other criteria of fiber selection like the defaults options we uses (I do not see which one ... ???)



Example of visualization (in blue (with transparency) the cortical associative roi in red the fdt path in green the seed_to_associative volume)

As you can see the associative class take the major part of the thalamus.

Select the biggest volume in fslview
and choose in fslview Menu : Tools-> image histogram

This may be due to a bias of the chosen cortical ROI. If we want to compare the relative pattern of each seed_to* volume (instead of a quantitative comparison) we may want to normalize the seed_to* volume. A basic normalization is to divide by the mean. Doing so will reinforce the local maximum of each seed and give more importance to territory that are less globally connected. (not sure if this is a publish method, but it make sense to me, and here we can try ...)

We want to divide each seed by it mean value. we use mrstats to compute the mean (you can also

use fslstats)

```
for k in `ls seed*`; do mrstats $k --mask  
../../../../anat_done/first/rdti_rthalamus.nii.gz; done
```

the -mask option compute the mean only in the mask.

Make the previous command more quiet, to get only one number (the mean)

```
mrstats seeds_to_*temporal.nii.gz --mask  
../../../../anat_done/first/rdti_rthalamus.nii.gz -output mean -quiet  
36540
```

now we want to divide the seed_to*temporal volume by 36540, using mrcalc the command is :

```
mrcalc seeds_to_*temporal.nii.gz 36540 -div norm_seeds_to_temporal.nii.gz
```

in bash, if you put a command in special quote `` this will replace it by its results. So the previous division can be written as

```
mrcalc seeds_to_*temporal.nii.gz `mrstats seeds_to_*temporal.nii.gz  
--mask ../../../../anat_done/first/rdti_rthalamus.nii.gz -output mean -quiet` -div  
norm_seeds_to_temporal.nii.gz
```

This way we can make the command easily in a for loop for all seeds :

```
for k in `ls seed*`; do mrcalc $k `mrstats $k --mask  
../../../../anat_done/first/rdti_rthalamus.nii.gz -output mean -quiet` -div  
norm_$k; done  
ls -ltr
```

check the results have a mean of 1

```
for k in `ls norm_seed*`; do mrstats $k --mask  
../../../../anat_done/first/rdti_rthalamus.nii.gz ; done
```

Now we have the new seed_to_target we can run the classification command :

```
find_the_biggest norm_seeds_to_c1_* norm_big  
compare the reusults
```

Connectivity-based segmentation in the anat space

Perform the same analysis but this time using seed and target in the anat space.

So change the seed and the targets list (and the output directory) .

You also need to select

Seed spaces is not diffusion

```
select the Seed to diff transform : /anat_done/first/T1_to_DTI.mat
```

the T1_to_DTI.mat has been obtain previously with the flirt command. This tell fsl the linear transformation matrix to go from the seed space (T1 space) to the DTI space. This could be deduce automatically from the header (this is what we did with the flirt command) because we did the coregistration with spm.

Run the command to look at the probtrac.log file and see the probtracx2 command line, but cancel the run, it will take too long (there are 4 times more voxel to seed) .

Redoo the same visualisation and same find_the_biggest command. The results are in
/probtrac_done/classif_thalamus_anat

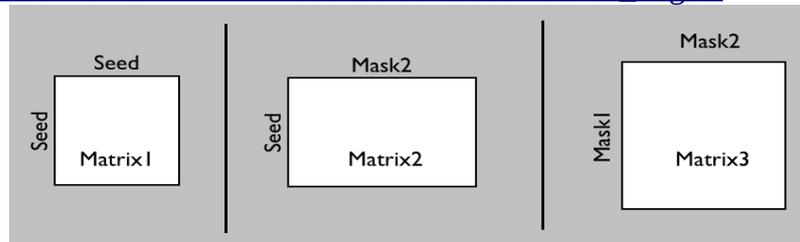
Blind classification : seed to white matter

We want to do the same but without target apriori. We want to classify the thalamus according the connectivity profile of each voxel.

We will use the matrix option of probtrackx (in the Option tab of probtrackx view). For our case we are interested by the -Matrix2 option

I copy paste the help from

http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FDT/UserGuide#Classification_targets



“Matrix Options The following set of options can be used to store a connectivity matrix between (1) all seed points and all other seed points, or (2) all seed points and all points in a target mask, or (3) all pairs of points in a target mask (or a pair of target masks). All masks used here **MUST BE IN SEED SPACE**. These options can be used separately or in conjunction.

Matrix1 and Matrix2 store the number of samples (potentially modulated by distance when distance correction is used) between the rows (seed points) and the columns of the matrix.

Typical uses of Matrix options:

Matrix1: The seed mask(s) can represent grey matter, so this would be all GM to GM connectivity.

Matrix2: The seed can be a grey matter region, and mask2 the rest of the brain (can be low res). The results can then be used for blind classification of the seed mask.

Matrix3: The 'target' masks can be the whole of grey matter, and the seed mask white matter. This option allows more sensitive reconstruction of grey-to-grey connectivity as pathways are seeded from all their constituting locations, rather than just their end-points as in Matrix1.”

We choose for Mask2 a mask of white matter in the DTI_space. So each line of Matrix2 represent a connectivity profile of a voxel seed (ie for each point of the white matter how many fiber go through). You can reslice to DTI space the c2*nii volume we get from spm segmentation of anat or directly segment the B0_mean.

```
Reslice the c2 mask and  
run the probtrackx with Matrix2 option. (it should take the same time as  
previously) since we continue seeding from the same seed
```

You find the result in DWI/probtrac_done/probtrac_rthalamus_matrix2_bin_white_pd

This command will create a fdt_matrix2.dot that you can clustered in octave

Run octave and use the following code

```
% Load Matrix2  
x=load('fdt_matrix2.dot');  
M=full(spconvert(x));  
% Calculate cross-correlation
```

```

CC = corr(M') %in matlab fsl documentation suggest 1+corrcoef(M');
whos CC
% this represent the correlation matrix of each seed voxel compared compared
to each seed voxel
% Do kmeans with k clusters
k=6
idx = kmeans(CC,k); % k is the number of clusters
% Load coordinate information to save results
addpath([getenv('FSLDIR') '/etc/matlab']);
[mask,~,scales] = read_avw('fdt_paths');
mask = 0*mask;
size(mask) % to is the volume dimension
coord = load('coords_for_fdt_matrix2')+1;
ind = sub2ind(size(mask),coord(:,1),coord(:,2),coord(:,3));
[~,~,j] = unique(idx);
mask(ind) = j;
save_avw(mask,'clusters','i',scales);
%previous nifti write does not handel the header, so a quick hack
unix('fslcpgeom fdt_paths clusters')

```

You can run those line with different values of k, just change the output name “cluster” compare the result obtain from target classification.

We made the same but seeding in anat space “probtrac_anat_thalamus_matrix2” subdirectory

Seed to target Classification with mrtrix

... TO BE completed ...

The data are in DWI_mrtrix. There is 2 version of mrtrix : mrtrix_single_mst and mrtrix_scale_mst which correspond to 2 different processing of the data

for mrtrix_single_mst we use the 4D_dwieddycor.nii.gz but for the other with use m4D_dwieddycor.nii.gz with is corrected for bias field.

For both with use the same processing except the last point.

```

dwi2response dhollander 4D_dwieddycor.nii.gz response_wm.txt response_gm.txt
response_csf.txt -grad grad.b -mask mask_mrtrix.nii.gz -voxels sf.nii.gz
-nthreads 1

```

```

dwi2fod msmt_csd 4D_dwieddycor.nii.gz response_wm.txt CSD_wm.nii.gz
response_gm.txt CSD_gm.nii.gz response_csf.txt CSD_csf.nii.gz -mask
mask_mrtrix.nii.gz -grad grad.b -nthreads 1

```

```

mrconvert CSD_wm.nii.gz wm_density.nii.gz -coord 3 0

```

```

mrcat wm_density.nii.gz CSD_gm.nii.gz CSD_csf.nii.gz tissueRGB.nii.gz -axis 3

```

```

tckgen -nthreads 24 -seed_image ../seg_spm/c2B0_mean_4D_dwieddycor.nii -number
10000000 -algorithm iFOD2 -grad grad.b -maxnum 10000000 CSD_wm.nii.gz
seed_c2B0_10M.tck

```

for the mrtrix_scale_mst we add a sift2 command to further correct from tractography bias.

```
sift2 seed_c2B0Gad.tck seed_c2B0Gad_weights
```

filter the track and put the results in a new subdirectory "classif_10M_fiber"

```
cd DWI_mrtrix/mrtrix_single_mst
mkdir classif_10M_fiber
for k in `cd ../../roi_done; ls c1*`; do echo tckedit seed_c2B0_10M.tck
classif_10M_fiber/seed_to_${k%.*}.tck -include ../../roi_done/$k
--include ../../anat_done/first/rthalamus.nii.gz; done
#reduo without the echo
```

convert it to probability volume (voxel value=number of fiber that cross the voxel)

```
cd classif_10M_fiber
ls -ltr
for k in `ls seed_to*tck`; do echo tckmap $k -template
../../../../anat_done/first/rthalamus.nii.gz ${k%.*}_prob.nii; done
#reduo without the echo
ls -ltr
```

multiply by the thalamus, to have the seed_to_target equivalent of fsl

```
for k in `ls seed_to*prob.nii`; do echo mrcalc $k
../../../../anat_done/first/rthalamus.nii.gz -mult thal_${k}_prob.nii; done
#reduo without the echo
ls -ltr
find_the_biggest thal_seed_to_c1_ranatwaal_* biggest
```

```
for k in `ls thal_seed*`; do mrcalc $k `mrstats $k --mask
../../../../anat_done/first/rthalamus.nii.gz -output mean -quiet` -div norm_$k;
done
ls -ltr
find_the_biggest norm_thal_seed_to_c1_ranatwaal_* norm_biggest
```

Look at the filter tract in mrview (option-> tractography). The advantage of mrtrix is that we have the view of each specific tracks.