

Table of Contents	
data download and setup.....	1
load metabolite structure	1
Inputs and flags for metabolite mapping.....	1
Get noise decoorelation matrix	1
assemble inputs and processing flags.....	1
Process data.....	1
Plotting.....	1
Data structure of Metcon_ME object.....	1
Overlay plot with anatomy.....	2
Other miscallaneous methods which can be useful.....	2
Debug (only for 'pinv' (linear) fit mode).....	3

Demo 04: 3D ME invivo data data

data download and setup

add paths and set up data paths. Download in invivo data (3.8 GB "sub-01-DMI.tar.gz") from [zenodo.org](https://zenodo.org/record/14652737) (DOI: 10.5281/zenodo.14652737).

```
% add all dependencies
addpath(genpath('/ptmp/pvalsala/Packages/mapVBVD'))
addpath(genpath('/ptmp/pvalsala/Packages/DeuteMetCon'))

% setup data folder
sn='/ptmp/pvalsala/deuterium/dataForPublication/sub-01-DMI';
dirst_me=dir(fullfile(sn,"*trufi_5E*.dat"));
dirst_noise=dir(fullfile(sn,"*trufi*noise*.dat"));
```

load metabolite structure

The chemical shifts, measured relaxation times and labels of all four metabolites in phantom is organised in a array of struct

```
% second argument is the frequency offset of water in Hz
metabolites=getMetaboliteStruct('invivo',0);
```

Inputs and flags for metabolite mapping

All functions and data required for all data processing steps like image reconstruciton, coil combination and spectral seperation were encapsualted in a single class `MetCon_CSI.m`. All inputs and flags except the raw data file is a name-value pair as described below.

%	name	description	default	possible options
%	metabolites	struct array with definition of metabolites	[]	see getMetaboliteStruct.m function
%	fm	1H fieldmap in rad/s	[]	3D numeric matrix or 'IDEAL'
%	csm	coil maps	[]	3D numeric matrix
%	mask	mask for spectral separation	[]	3D logical matrix , scalar percentile (1-100) threshold
%	doDenosing	SVD denoising	0	scalar No of components, -1 for debug
%	Solver	spectral separation method	'IDEAL'	{'phaseonly','pinv','IDEAL','IDEAL-modes','AMARES','LorentzFit'}
%				'phaseonly'- linear method with only phase evolution
%				'pinv'- linear method with full signal model
%				'IDEAL'- iterative IDEAL algorithm
%				'IDEAL-modes'-IDEAL algorithm for phase cycled data
%				'AMARES'- AMARES spectral fitting
%				'LorentzFit'- lorentzian spectral fitting
%	parfor	flag to use parfor	true	boolean
%	doZeroPad	zero pad factor	[1 1 1 0]	positive scalar array [3 physical axis x 1 time]
%	'doSmoothFM','maxit'	IDEAL flags: fieldmap smooth factor and maximum iterations	1,10	scalar(+ve: gaussian, -ve: median),postive scalar
%	doPhaseCorr	phase correction mode	'none'	{'none','Manual','Burg'}
%	'CoilSel','PCSel','EchoSel'	arrays to picks some of coils, time points and phasecycles.	1:max()	positive integer array
%	doNoiseDecorr	flag to perform noise decorrelation	true	boolean

Get noise decoorelation matrix

trufi sequence acquire noise data only when parallel imaging is enabled. Therefore, we acquire noise scan with 0 flip angle seperately.

```
twix_noise=mapVBVD(fullfile(sn,dirst_noise(1).name),'rmos');
```

```
Software version: VD (1?)
Reader version: 1660732089 (UTC: 17-Aug-2022 10:28:09)
Scan 1/1, read all mdhs:
    40.4 MB read in    7 s
```

```
[D_noise,D_image,noise_info]=CalcNoiseDecorrMat(twix_noise);
%D_image is the noise correlation from image data measured with 0 FA
```

assemble inputs and processing flags

```
ME_setting={ 'NoiseDecorr',D_image,'mask',[], 'metabolites',metabolites,...
             'doPhaseCorr',true,'doZeropad',[1 1 1]*0.5,'parfor',true,'fm',[], 'Solver', 'IDEAL-modes'};
```

```
%
% 'fm','IDEAL' : estimate field map from the averaged phasecycle volume using IDEAL algorithm
% 'Solver','pinv' : linear fit (least square fit of full signal model)
```

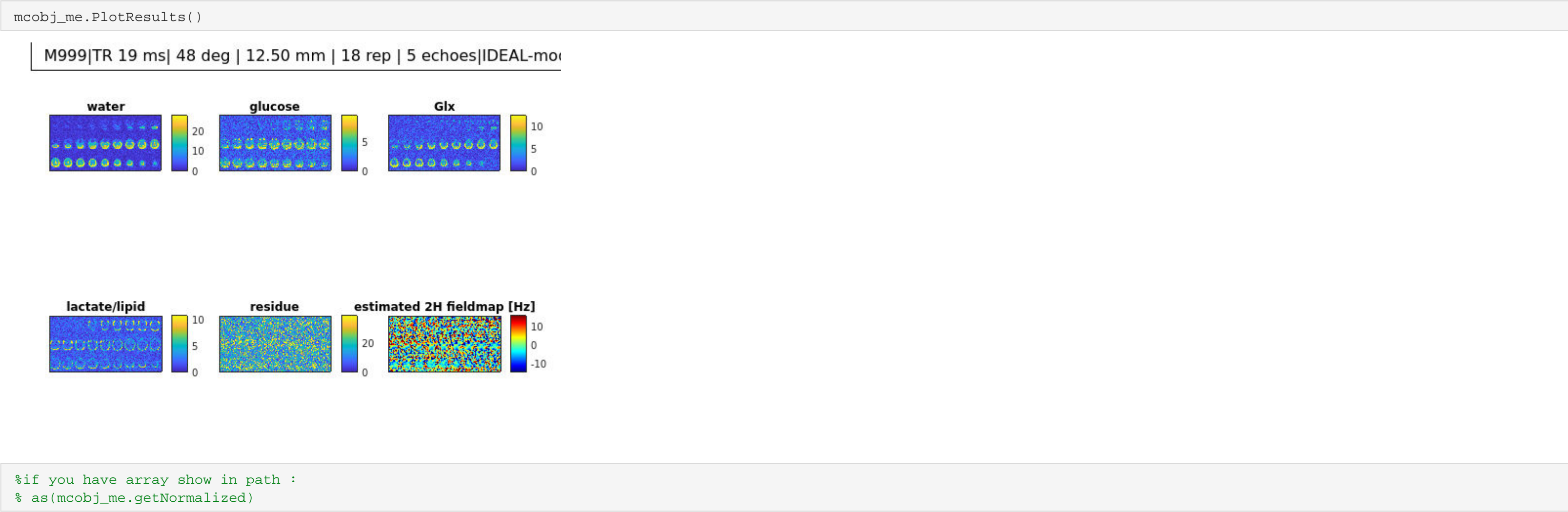
Process data

```
ME_filename=fullfile(sn,dirst_me(end).name);
mcobj_me=MetCon_ME(ME_filename,ME_setting{:});
```

```
Software version: VD (1?)
Reader version: 1660732089 (UTC: 17-Aug-2022 10:28:09)
Scan 1/1, read all mdhs:
    856.4 MB read in    9 s
starting reco
reco time =   19.1 s
estimating field map(1/2)
Starting parallel pool (parpool) using the 'Processes' profile ...
Connected to the parallel pool (number of workers: 32).
estimating metabolities(2/2)
Performed pixel shift along read: (0.0,-1.9,-4.9 ,-6.9) mm
Metabolite mapping time =   64.8 s
```

Plotting

After image reconstruciton and spectral seperation, metbolite amplitudes are store in **mcobj_me.Metcon**. The 4D Metcon matrix and 2H field map estimated by IDEAL algorithm can be quickly visualized with `PlotResults` method.



Data structure of Metcon_ME object

1x1

MetCon_ME

Property	Value
FieldMap	48x64x32 double
mask	48x64x32 logical
twix	1x1 struct
DMIPara	1x1 struct
flags	1x1 struct
filename	/ptmp/pvalsala/deuterium/dataForPublication/phantom-DM/MID0...
metabolites	1x4 struct
sig	6-D complex single
img	6-D complex single
coilSens	4-D complex single
coilNormMat	48x64x32 single
SolverObj	[]
Metcon	4-D complex double
Experimental	1x1 struct
D	10x10 complex single

Where

- DMIPara are the important sequence parameters parsed from twix with getDMIPara.m fucntion
- flags contains all the processing flags
- sig - averaged signal [CHA x LIN x COL x PAR x ECO x REP]
- img - reconstructed image [CHA x Phase x Read xSlice x echo x PC]
- Metcon - Metabolite amplitudes [CHA x 3 physical dimension x time]
- Experimental- contains all experimental outputs of processing (fieldmap, residue, other fit parameters, fit quality)
- D- noise decorrelation matrix.

Overlay plot with anatomy

```
%add SPM12
addpath('/ptmp/pvalsala/Packages/spm12/')

anat_tra=fullfile(sn,'anat/brain_tra.nii');

% export NIFTI volumes of all outputs (average image, metabolite amplitude in SNR unit and mM)
pn=fullfile(sn,'proc',sprintf('ME_%s',datetime('today','Format','yyyyMMdd')));
mkdir(pn);cd(pn);

Warning: Directory already exists.

niiFileName=mcobj_me.WriteImages(pn);
% reslice the metabolite amplitude to anatomy!
resliced_metcon=myspm_reslice(anat_tra,dir(fullfile(pn,'Metcon_SNR_*trufi*.nii')),'nearest','rt');

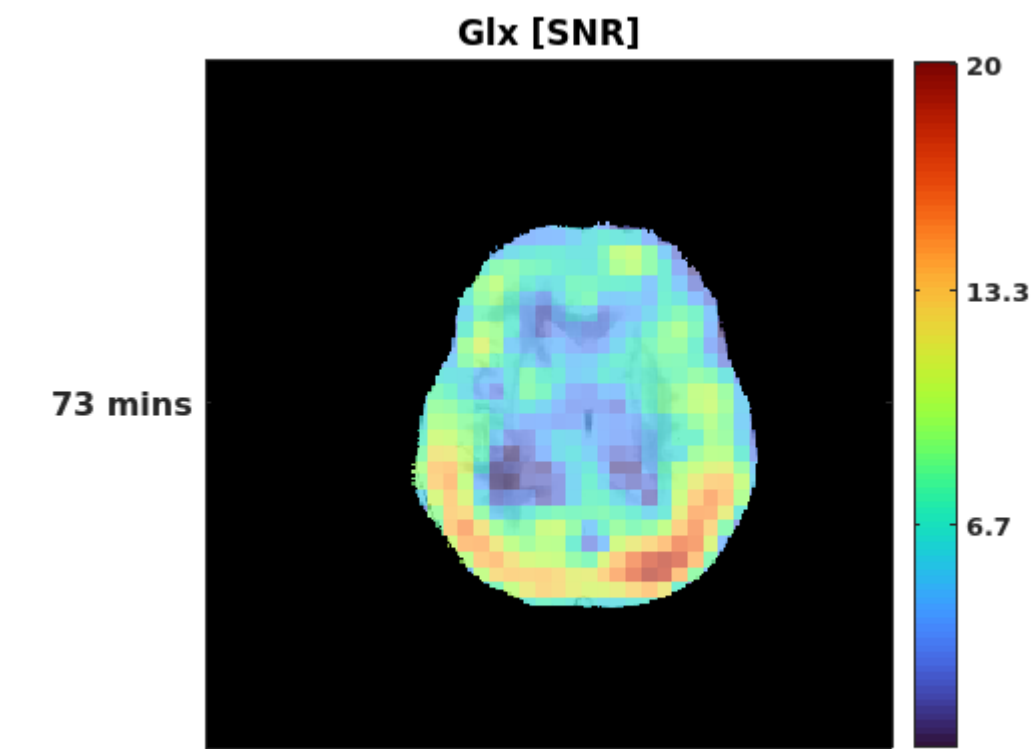
reslicing 1 volumes: Metcon_SNR_m00999_pvrh_trufi_5E_18PC_12P5mm_FA50_s4_r180_IDEAL-modes.nii

-----
15-Jan-2025 22:43:45 - Running job #1
-----
15-Jan-2025 22:43:45 - Running 'Coregister: Reslice'

SPM12: spm_reslice (v7141)                22:43:45 - 15/01/2025
=====
Completed                               : 22:43:46 - 15/01/2025
15-Jan-2025 22:43:46 - Done      'Coregister: Reslice'
15-Jan-2025 22:43:46 - Done

% Overlay plot
clim_glx=[0 20];
figure,
overlayplot(dir(anat_tra),dir(fullfile(pn,'rt*.nii')),'MetIdx',3,'SlcSel',15, ...
    'transform',@(x) flippermute(x,[2 1 3 4 5]),1),...
    'cax',clim_glx,'cmap','turbo');
title([metabolites(3).name,' [SNR]']) % 'MetIdx',3
yticklabels(sprintf('%d mins',mcobj_me.getMinutesAfterIntake('08:36')));

%make your own colorbar
cb_handle=colorbar;
cb_handle.Visible='off';
ax2 = axes('Position',cb_handle.Position);
imagesc(linspace(0,1,100)'),colormap(ax2,'turbo')
set(ax2,'YAxisLocation','right','FontSize',10,'FontWeight','bold','YDir','normal')
yticks(round(linspace(0,100,4)))
yticklabels(round(linspace(0,1,4)*clim_glx(2),1))
xticks([])
```



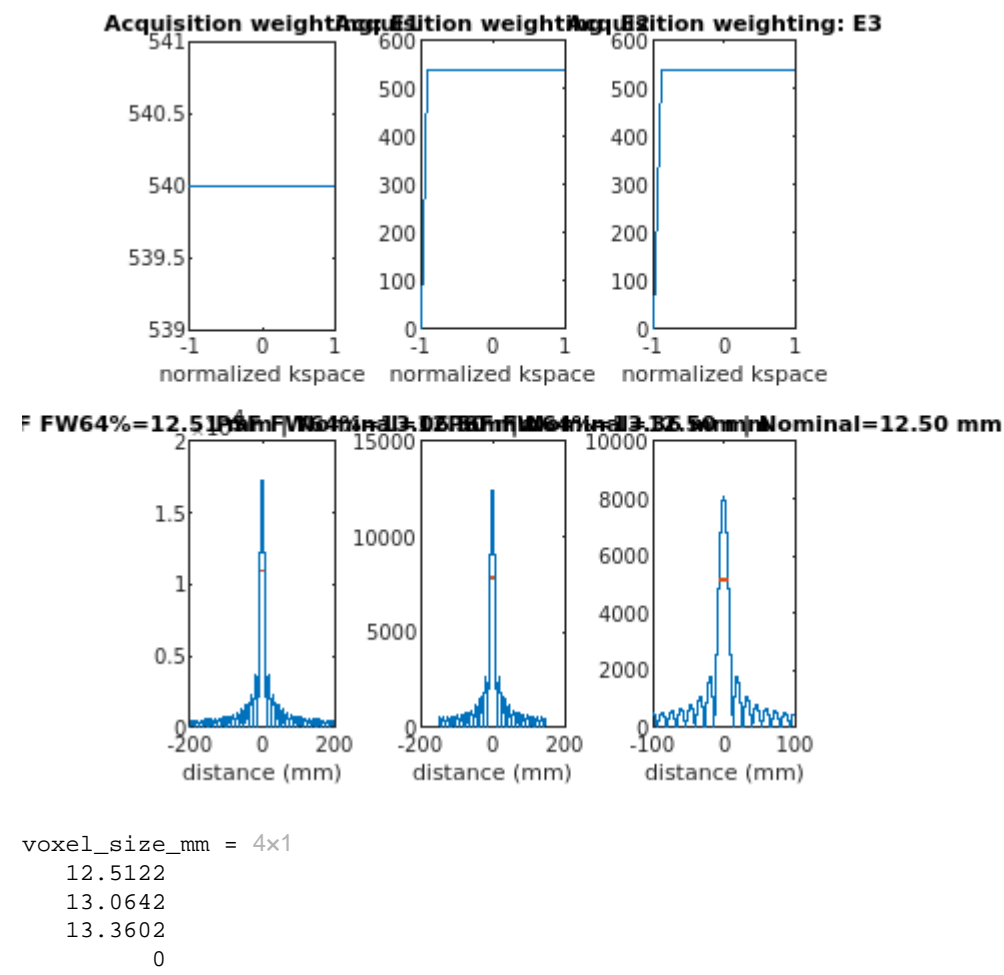
Other miscallaneous methods which can be useful

```
% The metabolite amplitudes can be normalized into SNR
metcon_SNR= mcobj_me.getNormalized();

% method and quantified into mM with 10 mM water reference using `getmM`
% fuction.
metcon_mM= mcobj_me.getmM();

% caculate measurement time after glucose intake (08:14) in mins
Intake_time_mins=mcobj_me.getMinutesAfterIntake('08:14');

%plot the k-space weighting and PSF to get realistic voxel size (FW64%)
voxel_size_mm= getPSF_CSI(mcobj_me.twix)
```



Debug (only for 'pinv' (linear) fit mode)

Useful for checking data at a particular voxel index.[34,38,22]is a gray matter voxel.

```
%prints metabolite amplitudes and plot fit and basis functions  
% mcobj_me.demoFit([34,38,22])
```