Chapter 1

MRSMatlab Manual

Thomas Kremer
University of Liege

Hadrien Michel
University of Liege

Mike Müller-Petke
Leibniz Institute of Applied Geophysics

This document provides the basics to understand and handle the Matlab based software MRSmatlab, that allows the processing, modelling and inversion of surface nuclear magnetic resonance (SNMR) data sets.

Contents

1 Introduction

This document was redacted as a manual describing the functioning of the Matlab toolbox MRSMatlab, which is designed for the processing and inversion of surface nuclear magnetic resonance data sets. Our goal is to provide the key informations allowing the reader to apprehend, be able to use, and understand the role of all features in MRSMatlab.

However, for a deeper understanding of the mathematics, physics and programming behind the interface of MRSMAtlab, we will generally refer the reader to some appropriate scientific publications, or to the code itself, but of course, never in a fully exhaustive way.

The toolbox MRSMatlab was initially presented to the scientific community by Müller-Petke, Braun, et al. (2016). It includes features that were developped by many different researchers. The main contributors are, in alphabet order:

- Ahmad Behroozmand
- Martina Braun
- Stephan Costabel
- Raphael Duglosch
- Thomas Günther
- Marian Hertrich
- Jochen Lehman-Horn
- Mike Müller-Petke
- Jan Walbrecker

This guide is largely based on the first "Documentation and User's Guide", that was redacted in 2016 by the previously quoted researchers, along with the publication of the main article. Our goal in this new version is to provide more detailed explanations and illustrations of the different tools provided by MRSMatlab.

2 Get it running

Installation:

- Run installMRSmatlab.m ! → This will create a shortcut in the matlab shortcut toolbar named MRSmatlab.

To start:

- Press the MRSmatlab button in the shortcut tab of Matlab. This will set all necessary paths within matlab, allowing you afterwards to launch and start using the different modules. Be aware however that pressing the MRS-Matlab button does not produce any output, and that nothing will appear on your screen, it is only an activation button.
- Then, type MRSSignalPro, MRSKernel, MRSModeling, MRSFit or MRSQT-Inversion in the Matlab command window. This will open each of the single interfaces.

3 Overview

The MRSMatlab toolbox is composed of 5 main modules.

1. **MRSSignalPro**: This module allows loading the raw data acquired on the field either with the GMR (Vista Clara) or Numis (Iris Instruments) and perform advanced signal processing operations. The processed data is saved in Matlab format as *.mrsd.
2. **MRSKernel**: In this module, the user can enter acquisition and earth model parameters and calculate the resulting 1D sensitivity function. The Kernel is then saved in Matlab format as *.mrsk.
3. **MRSModeling**: This module allows setting a synthetic 1D groundwater model, and calculate the associated synthetic data for a given sensitivity function.
4. **MRSFit**: The module MRSFit is used for post-processing of the data, that is, perform interpolation of NMR parameters before going to inversion.
5. **MRSQTInversion**: This module proposes different inversion schemes to inverse field or synthetic data.

In the following, we will try to describe thoroughly each of these modules, providing detailed explanations and illustrated examples.

4 MRSKernel

The module MRSKernel allows the user to compute the Kernel function associated with acquisition parameters that have been or will be used in the field. The Kernel can be seen as the sensitivity function and is needed to proceed through data (forward) modeling as well as through the inversion process. To compute it, one should proceed to the following steps:

1. Enter loop characteristics
2. Set optimal z-discretization
3. Set optimal pulse sequence
4. Change geomagnetic field parameters if available
5. Set the resistivity model
6. Compute the Kernel
7. Save the computed kernel (extension .mrsk)

Each of these steps requires to set parameters at a specific location of the MRSKernel window, which are indicated on Fig. 1, and detailed in the following.

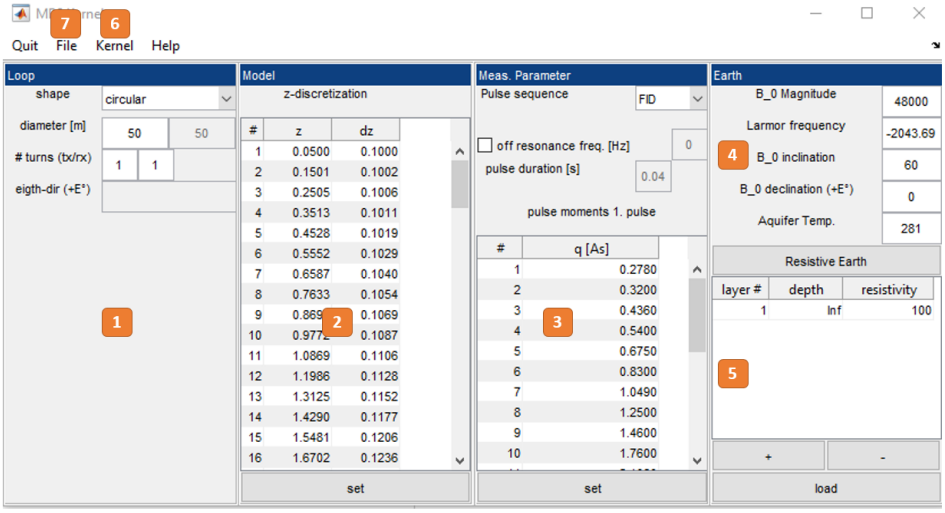
Note that many of these parameters values can be imported from field data files, if available. This is done using the command ***File*** $\rightarrow$ ***Import from field data***. If dealing with data from the GMR Vista Clara, the user must select the header file of the data set.

4.1 Enter loop characteristics

4.1.1 Shape

The user must indicate the shape of the loop that has been or that will be used as primary loop (for current injection) and as a receiver loop (for voltage measurement). This is done through the loop shape menu (Fig. 2). Note that except the for Inloop configuration, the shaped proposed implies a coincident-loop configuration, where the primary and receiver loop are the same.

The difference between circular or square loops is relatively marginal for large loops. The Kernel computation behind the scenes is always based on a circular loop and square loops are calculated based on the equivalent face. In the end, the



[H]

Figure 1: The MRSKernel window. The yellow numbered squares indicate the approximate location and general sequence order of the Kernel computation settings steps mentioned above.

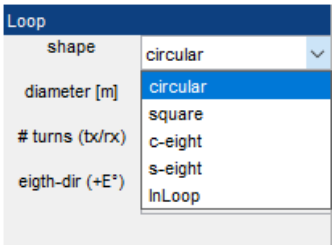


Figure 2: Image of the loop shape menu

choice between each geometry should primarily be made depending on the field acquisition conditions (obstacles density and distribution), as well as the tools available to set-up the loop (ropes or strings for circular loops, or right angle bezels for square loops).

Circular or square 8-shape loops will mostly be used when the EM noise conditions are expected or known to be difficult (see Trushkin (1994) for more details). Again, the use of square or circular shapes for these configurations will not affect significantly the quality and intensity of the NMR signal that will be acquired.

The InLoop configuration is relatively specific. It was implemented to compute the Kernel of the central-loop configuration (Behroozmand et al. 2016). The Inside

loop is the transmitter (Tx) and the Outside loop will be the receiver (Rx).

4.1.2 Diameter

The diameter value should be indicated in meters. For the 8 shape loops this value corresponds to the diameter of a single square or circle. For the Inloop configuration the diameter of each loop must be indicated separately.

In general, the diameter should be chosen according to the depth of investigation required (a rule of thumb can be: *diameter* $\approx$ *depth of investigation*).

4.1.3 Number of turns (tx/rx)

Indicate here the number of turns made on the primary loop (usually between 1 and 3). This value will vary with the actual diameter of the loop, which controls the inductance of the loop system. Small diameters will require to increase the number of turns so as to remain in the inductance range acceptable for SNMR acquisition with the GMR system. The GMR planner piece of software provided by Vista Clara can for example provide a good indication about the number of turns to be used.

The second number only needs to be indicated when the InLoop configuration is selected.

4.1.4 Eight-shape direction (+E°)

If using a figure-8-shape loop, the orientation of the loop in "East degrees" should be indicated here, as illustrated on the following scheme:

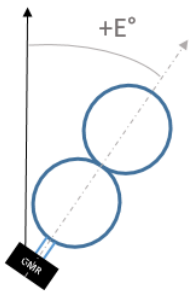


Figure 3: Illustration of the eight-shape loop orientation, in "East degrees"

This is important because, given a geomagnetic field with fixed inclination, the sensitivity function associated with the figure-8 configuration will change depending on the orientation of the loops (Trushkin 1994).

4.2 Set optimal z-discretization

Once the loop characteristics have been chosen, the user should set the discretization that will be used to compute the Kernel. To do so, press the ***set*** button, and the following window will open (Fig. 4).

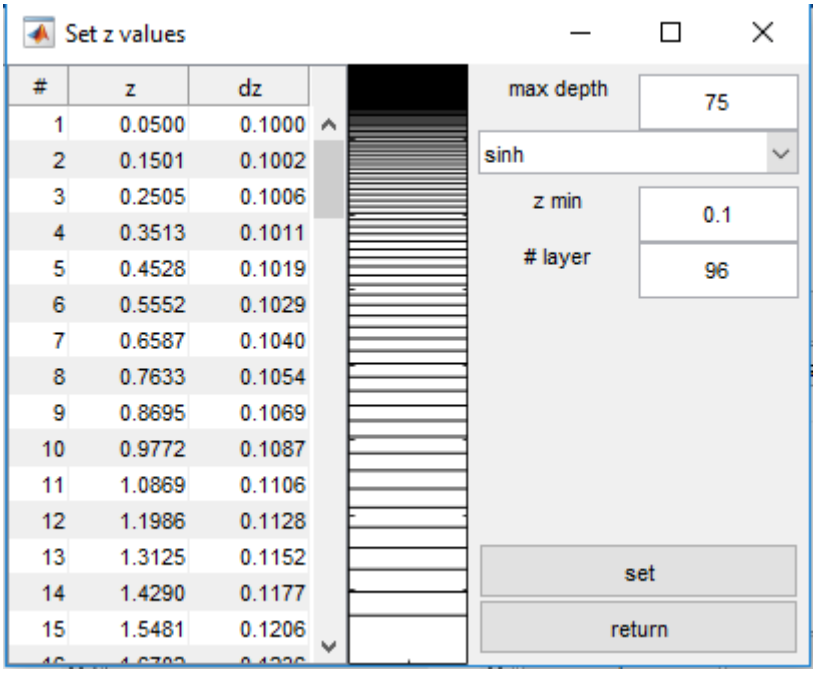
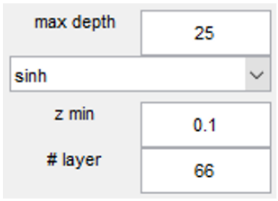
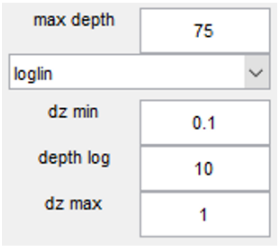


Figure 4: Image of the MRSMatlab window for z discretization

The user can specify the maximum depth of the discretization, the discretization function (hyperbolic sinus or logarithmic for shallow layers and then linear (loglin)). Depending on the function chosen, different parameters are to be set:



***z min*: minimum layer thickness**
***# layers* : number of layers**



***dz min*: minimum layer thickness**
***depth log*: depth (m) after which the function becomes linear**
***dz max*: maximum layer thickness**

Figure 5: Parameters setting menus

The discretization should be very dense in the shallow part of the model, and the maximum depth is advised to be at least 3 times the loop radius. Once these parameters have been set, the user need to press the ***set*** button to implement the changes required. Then the ***return*** button can be pressed to go back to the main MRSKernel interface.

4.3 Set optimal pulse sequence

- **Pulse sequence** Choose the desired pulse sequence, that is, define the NMR experiment for which you want to compute the Kernel function. Classic experiments use the FID pulse sequence. The T1 and T2 pulse sequences provide more advanced information. You can refer to A. A. Behroozmand, Keating & Auken 2015 for more detailed informations on the main pulse sequences used in SNMR surveys.

- **Off-resonance frequency [Hz]** The off-resonance frequency refers to the frequency shift between the ”exciting” magnetic field produced through the primary loop, and the *true* Larmor frequency associated with the geomagnetic field value. In an ideal case it should be zero, however geomagnetic field inhomogeneities and variations in time always lead to a small error. If the off-resonance frequency value is known thanks to previous signal processing schemes for example, it can be indicated here.

- **Set pulse moment sequence**

Upon pressing the ***set*** button, the window showed on Fig. ?? will open, where the user can choose the number of pulse moments composing the sequence (**# of q)**, the sequence distribution function (linear or logarithmic), and the value of the first and last pulse moment (**q min** and **q max**)

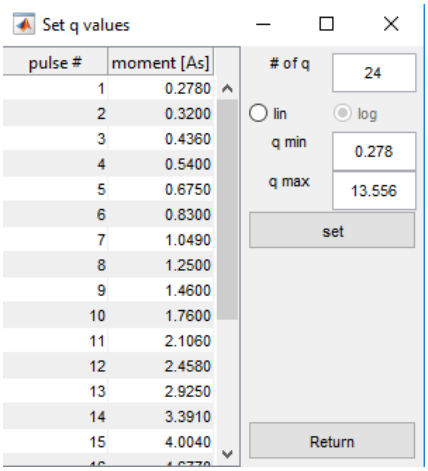


Figure 6: Pulse sequence setting menu

Again, if some parameters have been changed in this window, it is necessary to press the button ***set*** for them to be taken in account. After that the ***return*** button can be pressed to go back to the main MRSKernel interface.

4.4 Change geomagnetic field parameters

The MRS Kernel window provides default values for the characteristics of the geomagnetic field. However, when computing the kernel with the purpose of processing field data, one should enter the geomagnetic field characteristics of the survey location at the survey date.

For example, the user could refer to the geomagnetic model provided by the National Oceanic and Atmospheric Administration (NOAA), which can be reached here: https://www.ngdc.noaa.gov/geomag-web

The same kind of geomagnetic models could be used to try and approximate the geomagnetic field characteristics within a forward modeling study. However, it should be noted that these models are not precise enough to calibrate the Larmor frequency of the applied magnetic field, and should not be used to do so.

4.5 Set resistivity model

The ground resistivity will influence the Kernel shape and values and should be taken in account particularly when conductive grounds are present. The estimated conductivity model can be set by adding or removing layers (pressing the buttons + and -), whose depth (*lower interface* – in meters) and resistivity (Ohm.m) can be specified.

If the conductivity of the ground is totally unknown, the default model will be a homogenous layer of resistivity 100 Ohm.m

If the ground is globally resistive, that is, if all the layers composing the ground can be assumed to have a resistivity not significantly impacting on the B-field (as a rule of thumb resistivity can be neglected if higher than twice the loop diameter - see Braun & Yaramanci (2008)), a "Resistive Earth" model can be assumed. This choice can be implemented by clicking on the ***Resistive Earth*** button. Doing so will drastically reduce the time needed for the Kernel computation.

4.6 Compute the Kernel

To compute the sensitivity function, go to the menu ***Kernel*** then choose ***Make***. A progress window will open, displaying the elapsed computing time and the numpber of layers being computed. Once finished, the computed Kernel will be displayed on the figures.

4.7 Save Kernel

Go to the menu ***File*** then ***Save kernel as***. The Kernel will be automatically saved with a format .mrsk.

5 MRS Modeling

The module MRSModelling allows to perform forward modelling of SNMR data, that is, simulating the data that would be produced by a specific 1D groundwater distribution (thickness of aquifers, water content, relaxation time etc…), given some specific acquisition parameters (loop characteristics, pulse sequence etc…).

When appearing for the first time, the MRSModelling module will be composed of 4 different windows. At first, those windows will be pretty much empty, but information and results will show up as the modelling process progresses. To achieve modelling, the user needs to go through the following steps:

1. Load the Kernel file (.mrsk)

2. Set the expected groundwater distribution
3. Set the measurement parameters
4. Run the modelling
5. Save the modelled data

5.1 Load Kernel

Go to ***File*** => ***Load Kernel*** menu and select the appropriate .mrsk file

5.2 Set the expected groundwater distribution

Use the buttons + and − to add or remove layers, and specify the depth of the lower interface of each layer (in meters), the water content wc (%) and the T2 relaxation time (ms) within each layer, and if needed the ratio between the longitudinal and transversal relaxation times T1 and T2 (if planning to model specific pulse sequences which involve T1 measurements).

5.3 Set the measurement parameters

- *Pulse sequence*
- *Dead time* : with the GMR software usually 5 ms (default value)
- *SampleFreq* : sampling frequency of the simulated signal (real signals are sampled at 50 kHz with the GMR)
- *Record time* : default value is 0.5 s but actual field record are generally 1 s long.
- *Noise* : the synthetic mean noise level.

5.4 Run modelling

Press the ***Run modelling*** button

When finished, the three remaining windows will display graphs showing the input parameters and the resulting modelled data:

- The **model parameter graph** (Fig. 7) shows the Kernel function (left), the assumed resistivity distribution and the groundwater distribution within the subsurface, as specified on the first window.

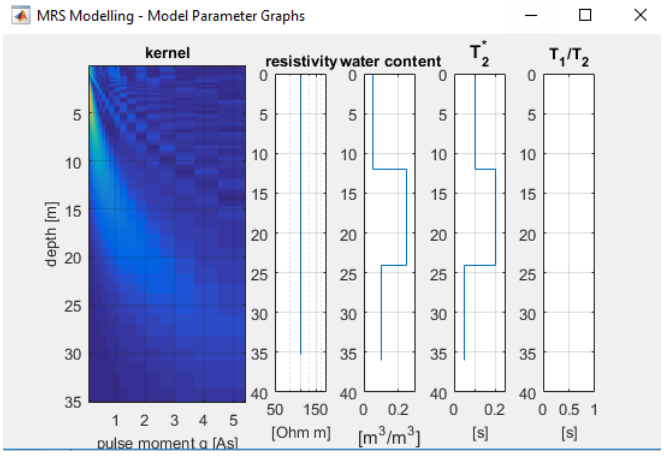


Figure 7: Model parameter window

- The **Data QT graph** (Fig. 8) shows the real parts and the imaginary parts of the complex signals obtained with each pulse moment. On the following screenshot the axis names are actually wrong, the x-axis is the time and the y-axis the pulse moment value. The color represents voltage amplitude.

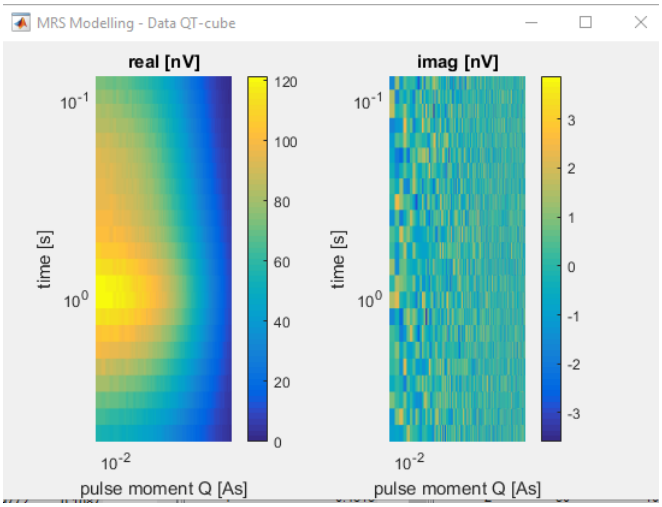


Figure 8: Real part (left) and Imaginary part (right) of the complex signals.

- The **Data Sounding graph** (Fig. 9) shows the simulated data under the form of sounding curves for each parameters: *initial amplitude (nV), apparent relaxation time (T2*/ms), and phase (rad).*

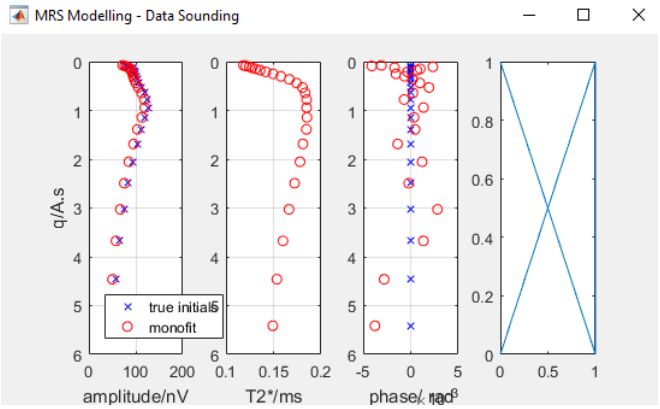


Figure 9: Real part (left) and Imaginary part (right) of the complex signals.

5.5 Save the modelled data

Go to ***File*** => ***Save Data***, and the data will be saved under the extension .mrsd

6 MRSSignalPro

The module MRSSignalPro allows processing several different signal processing operations on the acquired data. Once entering ”MRSSignalPro” in the Matlab command line, two windows will open: one for signal visualization entitled “data viewer”, and the second to command the signal processing operations. This task can be discretized in three different steps:

1. Load the data files
2. Proceed to the necessary or desired signal processing operations.
3. Save the processed data.

MRSSignalPro can load data from NUMIS, GMR and NMR Midi. Depending on the surface-NMR system used slight differences are present, e.g. phase-cycling is only present in GMR data.

6.1 Load the data files

6.1.1 GMR

Before starting, know that GMR uses phase-cycling to improve data quality. . Hence, only an even number of stacks should be loaded and processed. If this condition is not respected, a window indicating the problematic pulse moments will appear at the beginning of the stacking procedure (***save as stacked data*** button)

The first step to achieve is to load the .lvm files that contain the raw data acquired on the field. This can be done through the sequence of command ***File*** => ***Load raw data***. The user must then select the folder containing the .lvm files, and specify that the type of GMR data is “Raw data”.

Finally, the user is asked to open the GMR header file. This file is a text file which contains some information and parameters about how the study was conducted (e.g. voltage gain, loop information etc..). It must be named as the base name of the .lvm files to be read, that is, if the files are all named “stack_1.lvm, stack_2.lvm” etc..., the header file name must be “stack”.

Once it is selected, the information from all .lvm files (all stacks) present in the folder will be imported into the MRSSignalPro module, and upon the end of the importation process, signals will be displayed on the data viewer window.

6.1.2 NUMIS

Before starting, know that you need to the raw data (*.PRO) and not only the stacked data files. To load the data select the folder that contains the inp-files. This must be the folder that also contains the RawData folder with the *.PRO files. Once selected the data is loaded automatically and information from the inp file(s) are imported as well.

6.1.3 NMR MIDI

- to be decribed -

6.2 Data Viewer

The data viewer allows you to inspect the data in detail of every single record, processing steps done but also enables you to overview the complete dataset. In general the data is on display after a synchronous detection. Synchronous detection is done on the fly and only for display. The left side shows the selected single record (depending on what is selected in the command window) in time domain (real and imaginary part) as well as frequency domain. In the middle stacked data is shown for the selected pulse moment. The right most column shows the stacked data for all q in time domain (top) and frequency domain and allows for a good overview, especially in the frequency domain harmonic noise (vertical lines over all q) or spiky data (broadband increase over all frequencies) is quickly identified. Finally the bottom panel shows the data in frequency domain for current pulse-moment and give a noise estimate. Especially this noise estimate can help to evaluate the quality of any processing done later on.

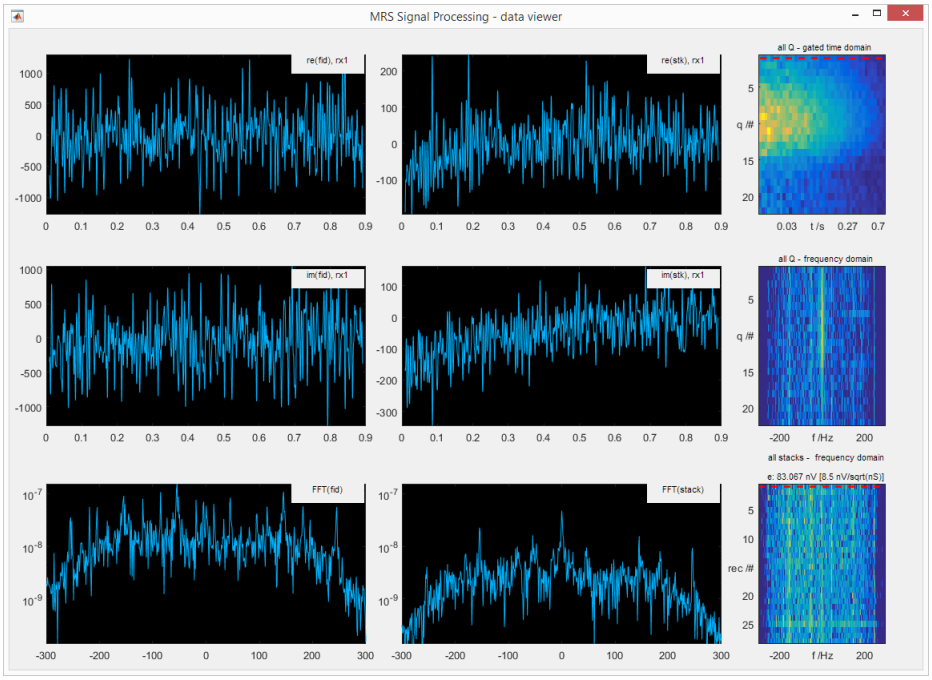


Figure 10: Data Viewer window

6.3 Proceed to signal processing operations

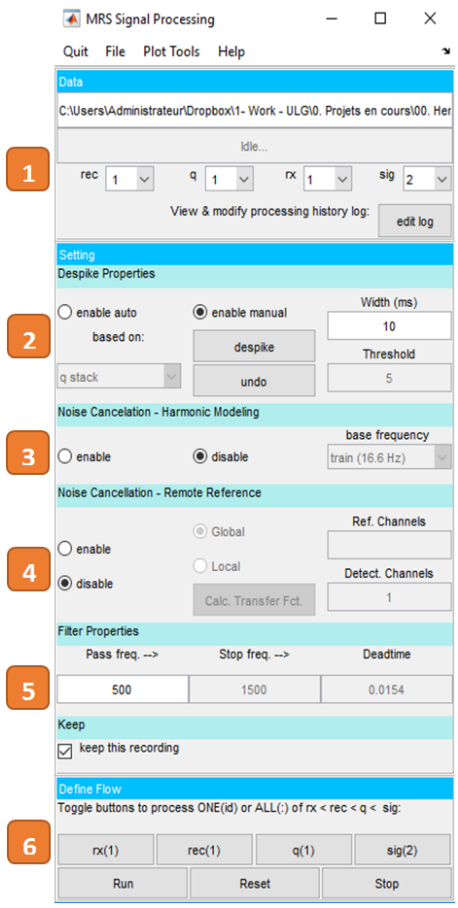


Figure 11: MRSSignalPro command window

This is the primary step which consists of selecting the appropriate signal processing operations that will be performed on the noisy data.

All these operations are implemented from the command window. This one is composed of three panels: the *Data* panel, the *Setting* panel and the *Define flow* panel.

The *Data* panel (1) is only used for signal visualization. The user can select the stack number (rec), the pulse moment number (q), the reception channel (rx) which can allow visualizing the noise on reference loops for example. The signal number (sig) should always be 2 when working with GMR data.

The *Setting* panel (2,3,4,5) allows to select the appropriate sequence of signal processing operations (detailed below)

The *Define flow* panel (6) allows selecting which signal will be processed. The user can either choose the select one signal (the one being visualized), or all by pressing the buttons rx, rec, q in which case symbols ":" will appear within the parenthesis.

6.3.1 Set the appropriate sequence of signal processing operations

The choice of the signal processing operations sequence is at the discretion of the user, and can depend on the number of data available (e.g reference loops or

not) and on a preliminary idea of what are the main components of the noise, for example thanks to visualization of the signals (e.g. are there spikes or not ? harmonics ?).

However, a general classic and complete sequence can be as follow:

1. Despiking (2)
2. Harmonic modelling (3)
3. Reference noise cancellation (if available) (4)
4. Despiking (2) if step 2 and 3 reveal new spikes (visualization check)

Whatever the sequence of operations that is chosen, you may process them one by one, and save the current state of the processing after each operation, using the command ***File => save current state***.

6.3.2 Description of the different signal processing operations

Despiking (2)

Spikes are very short (few ms) EM events with high amplitudes that can contaminate randomly the recorded signals. The procedure applied in MRSMatlab to remove them is presented in Costabel & Müller-Petke (2014). It basically consists of detecting spikes events through computation of standard deviations, and replace the signal within a defined time window by the stacked signal (using the other stacks), while excluding the detected spike.

In the despiking panel, the user can either choose to manually select the spike events which need to be removed, and set the width of the time window used for removal and replacement. In manual mode, the user presses the despike button, and then must manually place the cursor on the signal in the data viewer window, and click on it. This will result in the removal of the spike. If wrongly performed, the user can cancel the removal of the spike by clicking on the undo button.

Otherwise, automatic removal can also be selected, in which case the user must also precise the threshold that will constrain the detection procedure. The default value of 5 usually gives good results, but visual confirmation can help ensure that the threshold is optimal or not.

Manual removal is done by moving the different stacks and literally clicking on the spiky events that need removal. In automatic mode however, the despiking procedure must be launched by clicking on the Run button from the Define flow

panel (6), after having selected which records will be processed (after setting *rx*, *rec*, and *q*)

Harmonic modeling (3)

The physical and mathematical processes composing the harmonic modelling noise cancellation operation is based on Larsen, Dalgaard & Auken (2014) and implemented as described in Müller-Petke, Braun, et al. (2016) and references therein. Besically, a harmonic signal model is constructed that best fit the real signal being processed, and is then subtracted from it.

To apply this operation, the user must select the *enable* option, and then select the nominal fundamental frequency value that is expected to compose the harmonic noise: 16.66 Hz for train railway electric lines, 50 Hz for powerlines in some countries (all of Europe, and others), and 60 Hz for powerlines in other countries (All of North America, and others).

To launch the processing, the user must again press the ***Run*** button in the ***Define flow*** panel, after having selected the records to be processed.

Difficulties are present if one of the higher harmonics is close to the Larmor frequency. Cancelling this so-called co-frequency harmonic may result in cancelling NMR signal. Current research has presented approaches to avoid this e.g. Wang, Jiang & Müller-Petke (2018). This option is enables by ticking the ”remove coF” field.

Reference noise cancellation (4)

The physical and mathematical processes composing the reference noise cancellation process are described in Müller-Petke & Costabel (2014) and references therein. The basic principle is to use the noise signals recorded on reference loops which were set far enough from the primary acquisition loop, in order to remove from the primary signal an estimation of the noise components computed using transfer function between the two loops. MRSmatlab uses the frequency domain calculation of the transfer function.

After enabling this processing option, the user can either select the “Global” approach for the transfer computation, recommended when noise components appear stable with time, or the “Local” approach, rather recommended in unstable noise conditions. More details about these two approaches are given in Müller-Petke & Costabel (2014).

It is also necessary to specify the channel number that needs to be used as a reference channel. In many studies, several reference loops may have been displayed and chosing to use one channel alone or several together can produce different processing results. Trying different combinations on single stacks is

then highly recommended to determine the best reference channel(s). If a single channel is used, the user must indicate the channel number, if several are used, the channel numbers must be separated by one space (e.g: "2 3").

Filter properties (5) In this panel, the user can specify the bandpass filter width that is to be used within the processing operations (in Hz).

Keep recording (5) Un-checking this box will result in disregarding from the data set the signal currently displayed in the data viewer, that is disregarding the signal associated with one pulse moment within one stack. Upon doing so the data viewer graphs will darken. However, be aware when doing so that GMR data always requires an **even number of signals** to proceed to the stacking procedure. This means that if one removes a signal associated with one pulse moment from one stack, another signal associated with the same pulse moment, but from another stack must be removed too, so as to always have an even number of signal stacked together for each pulse moment. If this condition is not respected, a window indicating the problematic pulse moment will appear at the beginning of the stacking procedure (***save as stacked data*** button)

6.4 Save the processed data

At this stage, the user should be able to see and assess if the sequence of processing operations has been effective enough to let the NMR signal appear on the stacked signals (see the data viewer window). Either an energy peak at the Larmor frequency (in the Fourier transform graph), or typical decreasing (or oscillating) envelop shapes (in the time domain graph) can attest of the efficiency of the processing, and hence demonstrate that the number of stacks used is sufficient to perform a full acquisition survey.

To proceed, the user must save the stacked data in the .mrsd format. This is done through the command : ***File*** => ***Save as Stacked Data***. The .mrsd file hence created will then be used within the MRSQTInversion module. Note that this step of stacking the data is mandatory before proceeding to the fitting and then inversion procedures, even if no particular signal processing operation has been applied.

7 MRSFit

Before proceeding to the inversion of this processed data set, it is first necessary to fit the data. In fact, using the QT Inversion mono-exponential fitting as pro-

vided by MRSFit is in principle not necessary. However, the mono-exponential fitting provides a potential frequency offset between the exitation frequency (that is used for the synchronous detection) and the Larmor frequency. This knowledge on off-resonance excitation is important for correct kernel calculation. Furthermore, the phase parameter is used for calculating the "rotated dataset".

Pressing the button "Fitting data" will open 3 windows. One is the MRSFit command window and the 2 others are for visualization of the fitting results.

Generally, this step requires very little action from the user:

1. *Load the data*: use the command File => Load and select the .mrsd file that was produced during the previous signal processing step (step 5).

2. *Fit the data:* Press the button Fit all located at the bottom right of the command window.

3. *Save the fitted data*: using the command File => Save, and choosing an appropriate name so as to know that this is fitted data (it will also be a .mrsd format file).

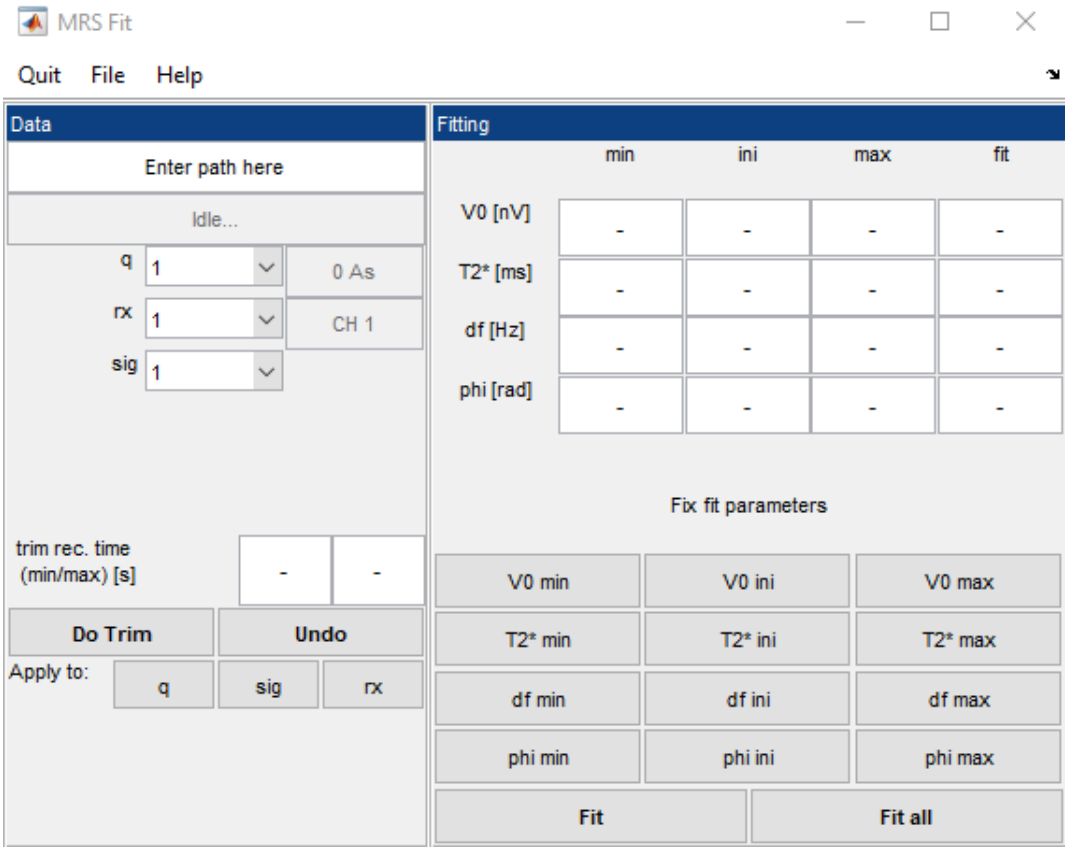


Figure 12: MRSFit command window

Only if the fitting operations seems largely unsuccessful, for example if the fitted data are not within the min or max values interval, the user could consider resetting these boundaries. This is done through the use of the ***Fix fit parameters*** buttons. However, the defaults parameters are generally reasonable values. So bad fitting results might simply mean that the NMR signal is not easily extractable, because of a high remaining noise level.

In this case, it might make sense to proceed back to the signal processing steps, while using a higher number of stacks so as to reduce further the residual noise level.

8 MRSQTInversion

After having fitted the data, the last step to be performed is the inversion of these signals. This procedure requires the module MRSQTInversion from MRSMatlab.

Three windows will open, the main command window 'MRS QT Inversion', from which the user will configure the inversion process, and two others windows for visualization of the data and of the inversion results (model).

Inversion processes can be cumbersome to parametrize optimally. If well versed in the resolution of inverse problems in the SNMR context, the user will find several ways of inverting data within this module, and detailed information in Müller-Petke, Braun, et al. (2016) and references therein.

Here we provide a few instructions that should help the user (even non-specialists) to proceed correct inversions of simulated or field data, without entering the details of the mathematical aspects behind. Basically, running through this process requires to follow the 4 steps below:

1. Load data
2. Load Kernel
3. Set inversion parameters
4. Run the inversion

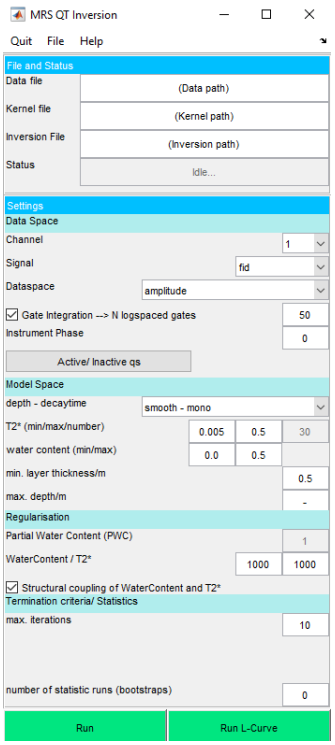


Figure 13: MRSQTInversion command window

1. *Load data*: use the command ***File*** => ***Load data*** to select the data previously fitted during step 6. The data visualization window will be updated.

2. *Load Kernel*: use the command ***File*** => ***Load Kernel*** and select the .mrsk file that was computed during step 1.

3. *Set inversion parameters*: at this stage the user could proceed directly to an inversion using the default parameters from the module, which are generally suited to classical cases, .

4. *Run inversion*: Running the inversion is done by pressing the button ***Run***, a few seconds will be needed for the iterative process to converge, and the visualization windows will be updated with the inversion results, as shown on the example below Fig. 14.

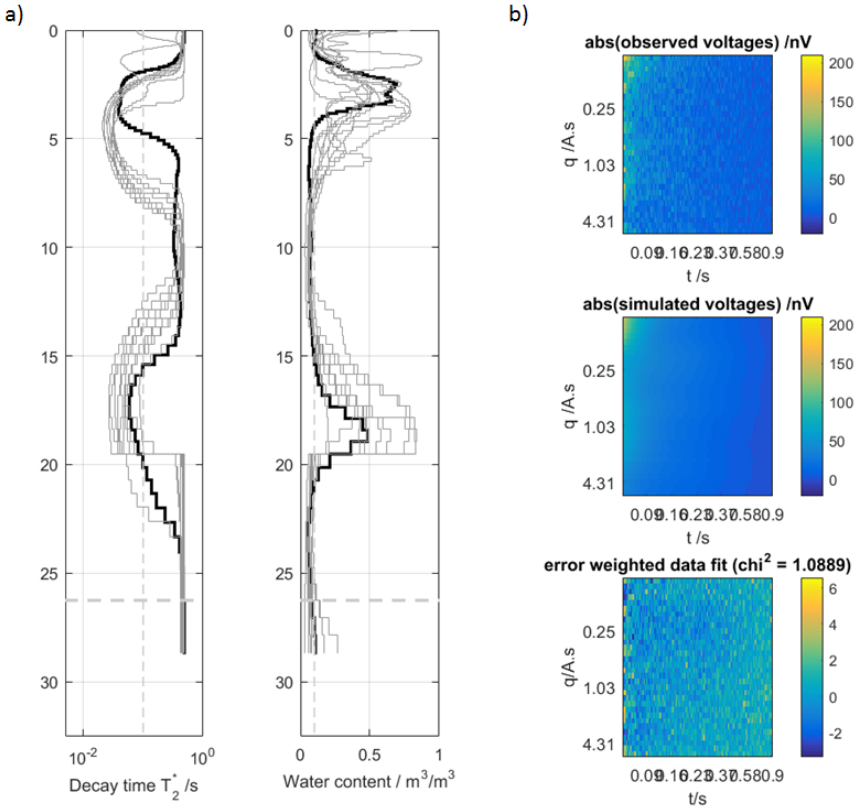


Figure 14: Inversion results example.

The 3 colored matrix on Fig. 14b) respectively show the data (observed voltage), the obtained model (simulated voltages), and the Chi2 error between these two distributions. The inversion can be considered correct if the Chi2 value is close to one (ideally between 0.9 and 1.1)

The two graphs on Fig. 14a) show the water content and relaxation times distribution obtained after inversion. If the user did not proceed to any other parameter modification, he should only observe the thick black curves. These curves are to be visually compared to the initial scenario used to simulate the NMR acquisition, so as to assess the quantity and quality of the retrieved information.

The thin grey lines can be obtained if the user set the option number of statistic runs to a non-zero value (10 is generally optimal). These curve allow to estimate the uncertainty on the inversion results. Basically if these curves all have similar shapes it means that the uncertainty on the results is low and the detected struc-

tures and water content values are likely to be good estimations. Otherwise, if the grey curves give very different shapes, it means that the data does not constrain sufficiently the inversion results, which, therefore, should not be trusted.

In the case presented on Fig. 14, one could consider that the uncertainty is quite acceptable. There are variations in the water content precise values, but globally the two structures are detected each time, indicating that they are very likely to be real.

More advanced parameters are available to better suit to the dataset that is inverted. Those settings are differentiates into four types:

- Data Space
- Model Space
- Regularization
- Termination criteria/Statistics

8.1 Data Space

The data space settings are used to select the correct way to represent the data in the inversion process.

- The *Channel* option enables the use of multichannel datasets by selecting the channel of interest corresponding to the kernel file previously selected.
- The *signal* option is used to select the type of data that are inverted (either the free induction decay - FID - or the Carr-Purcell-Meiboom-Gill pulse sequence - CPMG). The FID is the classically used pulse sequence but is unable to determine the value of T_2, contrary to the less common CPMG sequence. For further details on the this subject, the user is referred to A. A. Behroozmand, Keating & Auken (2015).
- The *dataspace* option lets the user define the best way to represent the dataset: amplitude, rotated complex or complex. Each possible option gives a different level of details on the dataset. The amplitude is the simplest one, reducing each complex value of the dataset by the corresponding amplitude, it is the default option and performs satisfyingly on most of the datasets. However, in some particular cases (when the subsurface is highly conductive, leading to an important phase shift for example), it is better to

use the complex value signal. In those cases, the use of the rotated complex or complex options are recommended. The rotated complex option is a lighter representation of the complex signal ?? while the full complex representation is the most complete representation of the signal, but also leading to heavier computations. Pay attention to the fact that the imaginary part of the signal is often one order of magnitude lower than the real part, although noise impacts both real and imaginary parts in the same way, leading to a lower signal-to-noise ratio in the imaginary part.

- The *gate integration* check-box lets the user choose between the full dataset or its reduced form using gate integration in time (log-space). To change the numbers of gates, use change the value in the text-box present in the same line (default value: 50). The use of this option will dramatically reduce the memory usage and computation times required to perform the inversion. Moreover, the effect of noise is generally reduced due to the random aspect of this latter (if the signal was properly processed).

- The *instrument phase* parameter allows to account for instrumentally introduced phase error.

- The *active/inactive qs* pushbutton opens a new GUI (Fig. 15) that enables the selection of a given set of pulse moments from the dataset. It can be useful to gain computation time in cases where some pulse moments are non-informative (no signal) and to reduce the impact of noise. The user can check/uncheck the checkbox behind each pulse moment to enable/disable. Once the selection is made press the *Save* pushbutton. If you want to discard the changes, press the "Cancel" pushbutton. The option is bugged (Matlab R2016a)!!! The Matlab function *findobj* (line 16 in the *getActiveQ*) returns an empty matrix in the case where the figure does not exist, leading to an error in the next line. The line 17 should be replaced by: if not(isempty(oldfigure)).

8.2 Model Space / Regularization / Termination criteria / Statistics

- The model space settings are used to tune the model space to the prior knowledge of the site/experimental design.

- The regularization settings are used to set optimal regularization parameters for the inversions.

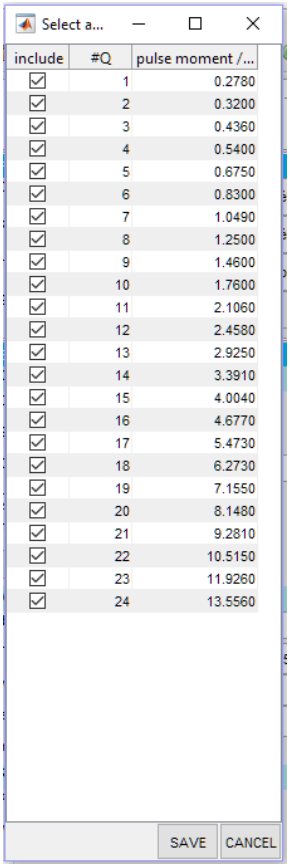


Figure 15: Example of the "Select active pulse moments windows".

- The termination criteria/statistics settings are used to determine criteria for the inversion process to stop.

As the different options that are enabled depends on the choice of the *depth-decaytime* option used, we will first describe the possibilities for this option. Then, for each choice of *depth-decaytime* possible, a further description of the other settings will be made.

The *depth-decaytime* option is used to define the used discretization of decay time with depth. Three options are available:

1. Smooth-Distribution: this is the first type of distribution that was introduced to MRSMatlab. It uses discretized relaxation time and water content

spaces. The inversion is therefore a linear process with partial water contents as only unknowns. The partial water content is the water content at a given depth associated to a given relaxation-time. The results are presented as in Fig. 16. Only the graph on the left is the result of the inversion, the two others (decay time and water content) are issued from post-processing routines. The associated water content is the sum of the partial water contents for a given depth and the represented relaxation time variation with depth is the corresponds to the value with the maximal partial water content.

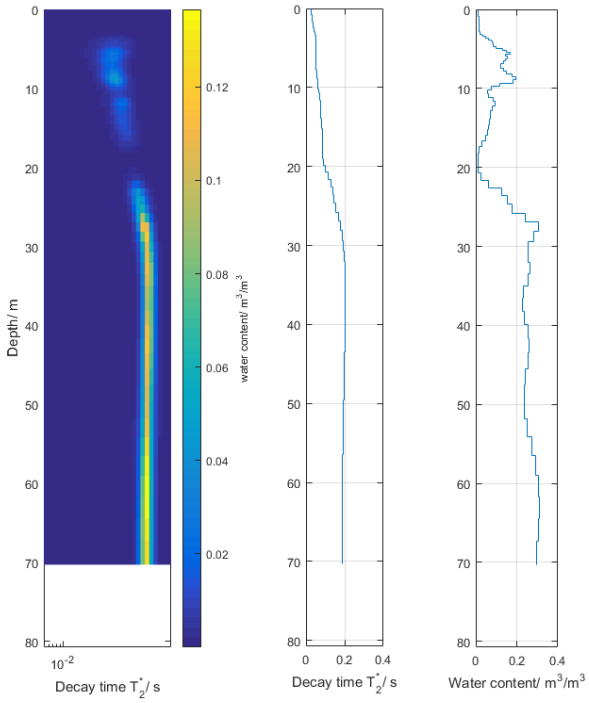


Figure 16: Example of smooth-distribution results.

2. Smooth-Mono: this is the default option for the inversion. In this case, the inversion predicts both the relaxation-time and the water content (not partial). Due to the non-linearity of the forward model in relaxation time (e.g., Müller-Petke & Yaramanci, 2010), the inversion process requires an update of the kernel function at each iteration. However, the process is less

memory demanding. The results are displayed as in Fig. 17. Both graphs are issued from the inversion process.

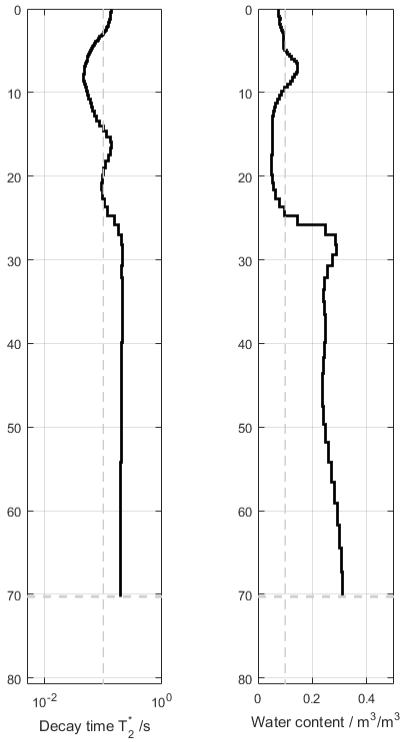


Figure 17: Example of smooth-mono results.

3. Block-Mono: this latter option uses a Genetic Algorithm instead of a classical inversion (Akca et al. 2014). The algorithm generates a set of models that are inside the search space and seeks for a set of possible models in the population. The results are displayed as in Fig. 18, which is very similar to the smooth-mono results with a blocky aspect.

8.2.1 Smooth-Distribution

The lower part of the settings panel is presented as in Fig. 19 when the user has set the *depth-decaytime* option to "Smooth-Distribution".

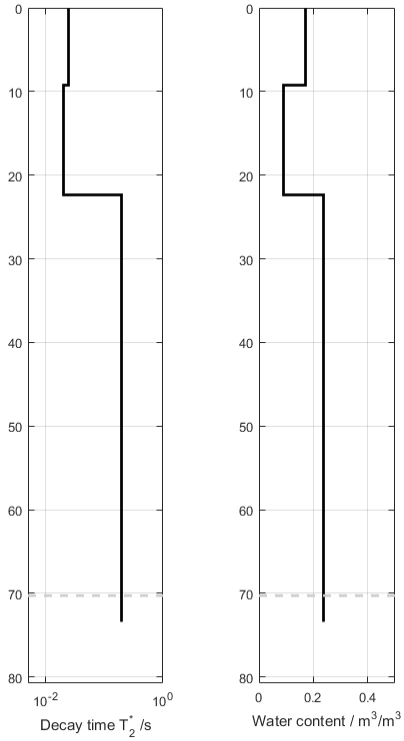


Figure 18: Example of block-mono results.

In the *Model Space* panel, the user can tune the remaining parameters to better suit to its case.

- The T_2^* *(min/max/number)* options are defining the used decay time distribution. The resulting discretization of the decay time will follow be a set of *number* decay times ranging from *min* to *max* with a logarithmic scale.
- The *water content* option is only available for the maximum value. The minimum value is automatically blocked to 0 due to the use of a logarithmic discretization of the water content. Changing this value will change the upper limit of the total water content in the inversion process.
- The *min. layer thickness (m)* option enables the user to choose for another depth discretization as the one proposed by default (corresponding to the

Figure 19: Lower part of the settings panel for the smooth-distribution depth-decaytime option.

Kernel file).

- The *max. depth (m)* option performs a modification of the maximal depth of the inverted models.

In the *Regularization* panel, the user can only change the regularization parameter for the *partial water content (PWC)*. This parameter is equal to 1 by default. Increasing the value of this parameter will lead to smoother variations in the resulting partial water content distribution and vice-versa.

The *Termination criteria* panel let the user set the stopping criteria for the iterative inversion process. Two criterion are proposed: a maximum number of iterations and a minimum model update. The process will automatically stop when one or both of the criteria are reached.

- The *max. iterations* option sets the maximum number of iterations that are authorized for the inversion. This does not guaranties any convergence of the iterative process but rather secures the computations by avoiding infinite loops. Increasing this value may lead to better results but will also increase computation time and vice-versa.

- The *min.model update* parameter is another way to stop the inversion process. It will stop the iterations when the computed model update will be

lower than this value. Increasing this value may lead to less accurate results but decreased computation times and vice-versa.

When using the *Smooth-Distribution* depth-decaytime option, the user cannot run the L-Curve in order to determine a value for the regularization parameter.

8.2.2 Smooth-Mono

The lower part of the settings panel is presented as in Fig. 20 when the user has set the *depth-decaytime* option to "Smooth-Mono". The options are more or less the same as for the previous case, with some exceptions. Nonetheless, we will describe all the options for those who choosed to pass the previous part.

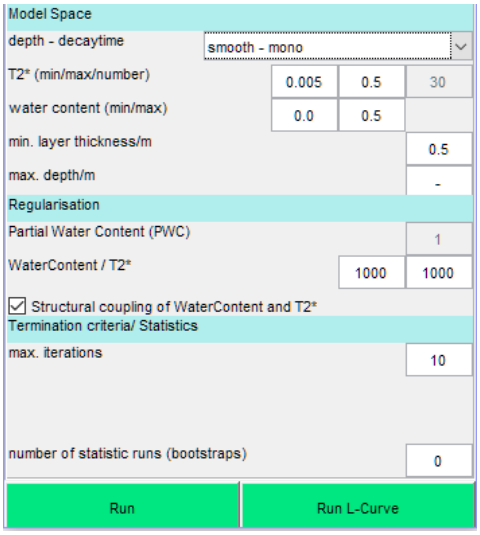


Figure 20: Lower part of the settings panel for the smooth-mono depth-decaytime option.

In the *Model Space* panel, the user can tune the remaining parameters to better suit to its case.

- The T_2^* *(min/max/number)* options are defining the lower and upper limits of the decay time distribution. The *number* option is grayed as it has no use in this case.
- The *water content* options are defining the lower and upper limits of the water content distribution.

- The *min. layer thickness (m)* option enables the user to choose for another depth discretization as the one proposed by default (corresponding to the Kernel file).
- The *max. depth (m)* option performs a modification of the maximal depth of the inverted models.

In the *Regularization* panel, different options are available to the user:

- The *Water content/T_2^** options enables changes to the regularization parameters applied to the water content (left) and decay time (right) distributions. The values can differ for both distributions.
- The *Structural coupling of Water Content and T_2^** check-box enables a decoupling of both distributions.

In order to determine an optimal regularization parameter, the L-curve criteria is proposed (*Run L-Curve* pushbutton at the bottom left of the window). This will run multiple inversions with different values for the regularization parameters and compute the norm on both the model and the data misfit. Then, a graph presenting the norm on the model versus on the data misfit for each regularization parameter value will be presented (Fig. 21). The L-Curve criteria suggests that the best trade-off between data misfit and the smoothness of the model is at the maximum of curvature of the L-curve (hence the name) (Aster, Borchers & Thurber 2013). For example, in the example presented in Fig. 21, the maximum of curvature is obtained for a value of the regularization parameter of about 500.

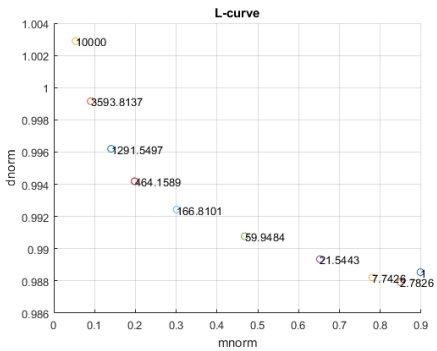


Figure 21: Example of L-curve.

The *Termination criteria/Statistics* panel let the user set the stopping criterion for the iterative inversion process. The only proposed stopping criterion is the maximum number of iterations. The *max. iterations* option sets the maximum number of iterations that are authorized for the inversion. This does not guaranties any convergence of the iterative process but rather secures the computations by avoiding infinite loops. Increasing this value may lead to better results but will also increase computation time and vice-versa. The default value of 10 iterations works properly on most of the datasets. However, if you observe in the "Iteration progress" graph (appearing during the inversion and disappearing just at the end of the process) that the the data misfit still decreases significantly at the last iterations, it may be worth trying to increase this parameter.

Finally, the *number of statistics runs (bootstrap)* option enables to use bootstrap inversions to asses roughly the uncertainty on the predicted model. The bootstrap algorithm uses subsamples of the original dataset that are all inverted and compared to estimate uncertainty (Hertrich 2008). The higher the number of bootstrap runs, the better the estimation of uncertainty. However, the method is highly CPU demanding and therefore a limited number of runs should be set (10 is often sufficient).

8.2.3 Block-Mono

The lower part of the settings panel is presented as in Fig. 22 when the user has set the *depth-decaytime* option to "Block-Mono". The options are more or less the same as for the previous case, with some exceptions. Nonetheless, we will describe all the options for those who choosed to pass the previous parts.

In the *Model Space* panel, the user can tune the remaining parameters to better suit to its case. In this type of interpretation, the definition of the model space will correspond to the explored space.

- The T_2^* *(min/max/number)* options are defining the lower and upper limits of the decay time distribution. The *number* option is grayed as it has no use in this case.
- The *water content (min/max)* options are defining the lower and upper limits of the water content distribution.
- The *layer thickness (min/max)* options are defining the lower and upper limits of the layer thickness (in meters).
- The *nLayer* option sets the number of layers in the models to a given number.

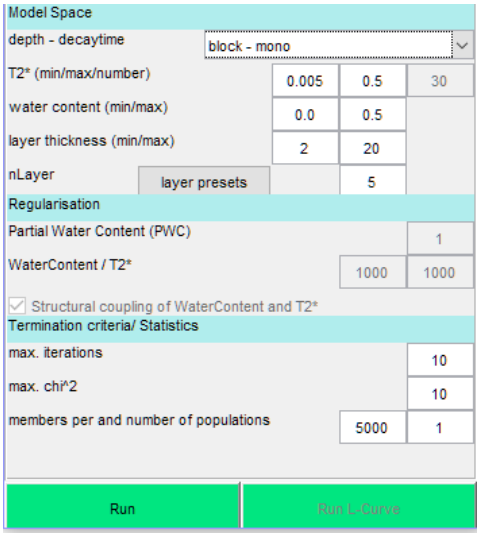


Figure 22: Lower part of the settings panel for the block-mono depth-decaytime option.

- The *Layer presets* pushbutton enables a better tunning of the search space. When pressed, a GUI appears (Fig. 23). In this table, the user can enter restrictions on the generated models based on prior knowledge of the site. Five options are proposed:
  - *Lower layer boundary*: Fixing a known boundary in the model (in meters)
  - *Water content*: Fixing a known value for the water content (in m^3/m^3)
  - T_2^*: Fixing a known value for the decay time (in seconds)
  - *Minimum thickness*: Fixing a minimum thickness different from the minimum thickness defined for all the layers in the previous steps
  - *Maximum thickness*: Fixing a maximum thickness different from the maximum thickness defined for all the layers in the previous steps

  Once the restrictions are properly set, press the *Save* pushbutton at the bottom of the GUI. To recover an unconstrained variable, type NaN at the location of the corresponding variable.

The *Regularization* panel is grayed out as no regularization parameter is used in the algorithm.

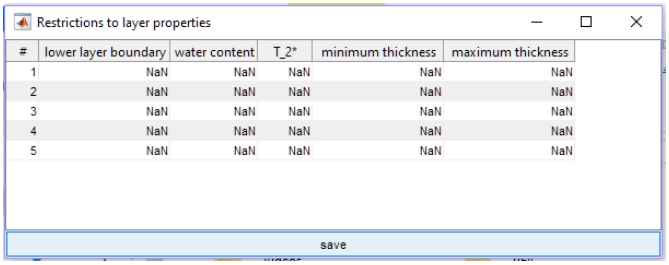


Figure 23: Layout of the ”Restriction to layer properties” GUI.

The *Termination criteria/Statistics* panel let the user set the stopping criterion for the iterative inversion process. The only proposed stopping criterion is the maximum number of iterations. The *max. iterations* option sets the maximum number of iterations that are authorized for the inversion. This does not guaranties any convergence of the iterative process but rather secures the computations by avoiding infinite loops. Increasing this value may lead to better results but will also increase computation time and vice-versa. The default value of 10 iterations works properly on most of the datasets. However, if you observe in the ”Iteration progress” graph (appearing during the inversion and disappearing just at the end of the process) that the the data misfit still decreases significantly at the last iterations, it may be worth trying to increase this parameter.

The *max. chi^2* parameters is used for graphical purposes and does not impact the genetic algorithm

Finally, the *members per and number of populations* parameters are there to define the number of models that are generated in one population (leading to one solution) (left) and the number of populations (or the number of resulting solutions) (right).

Again, the *Run L-curve* pushbutton is disabled as no regularization is applied during the genetic algorithm process.

8.3 Saving and manipulating the results

Once all the inversions that you wanted to perform are completed, you can save the obtained results to an *.mrsi file using the command ***File*** => ***Save data***. You then need to give a name to the file and choose its location.

To read the inversion file, you can open back MRSQTInversion through the Matlab Command Window and select in this order:

- File => Load Data

- File => Load Kernel
- File => Load Inversion

The last inversion that was performed will directly be displayed in the "Data" and "Model" figures. If multiple types of depth-decaytime distributions have been tested and saved in the inversion file, you can access those results by selecting the right depth-decaytime option and the corresponding results will be displayed (if they exist in the Inversion file).

Another way to access the inversion results is through the Command Window in Matlab.

```
idata = load('FileName.mrsi','-mat');
idata = idata.idata;
```

This will load the inversion file in a structure called idata. This structure is organized as follows:

```
idata =

       data: [1x1 struct]
     kernel: [1x1 struct]
         fn: []
    inv1Dqt: [1x1 struct]
       para: [1x1 struct]
```

- The idata.data structure contains the experimental data.
- The idata.kernel structure contains the sensitivity kernels
- The idata.fn structure contains
- The idata.inv1Dqt structure contains the inversion solutions
- The idata.para contains the last set of used parameters

All the results are contained in the idata.inv1Dqt structure.

```
>> results = idata.inv1Dqt

results =
```

```
          para: [1x1 struct]
    smoothMulti: [1x1 struct]
     smoothMono: [1x1 struct]
      blockMono: [1x1 struct]
```

As you can see, the results from the different types of depth-decaytime distributions are stored in different sub-structure:

- The *Smooth-Distribution* is stored in the results.smoothMulti structure
- The *Smooth-Mono* is stored in the results.smoothMono structure
- The *Block-Mono* is stored in the results.blockMono structure

The results.para structure is the same as the idata.para one.

```
>> results.smoothMono

ans =

           t: [1x49 double]
           z: [1x95 double]
           g: [24x95 double]
    solution: [1x1 struct]

>> results.smoothMulti

ans =

    decaySpecVec: [1x30 double]
               t: [1x49 double]
               z: [1x95 double]
               g: [24x95 double]
        solution: [1x1 struct]

>> results.blockMono

ans =

           t: [1x49 double]
           z: [1x96 double]
           g: [24x96 double]
    solution: [1x1 struct]
```

As can be observed, the structures containing the different solutions are very similar, with the only exception that the smoothMulti that also contains the decay-time discretization: decaySpecVec.

However, the solutions are expressed differently for each set *depth-decaytime* distributions:

- Smooth-Distribution: In this case, the model is a set of partial water contents. It is expressed as a vector called m_est. To display this solution, you need to follow the instructions:

```
M = reshape(idata.inv1Dqt.smoothMulti.solution(1).m_est, ...
    length(idata.inv1Dqt.smoothMulti.z), ...
    length(idata.inv1Dqt.smoothMulti.decaySpecVec)); % ...
    Changing to matrix form
figure('Units','Normalized','Position',[0.1 0.1 0.1 0.8]);
pcolor(idata.para.decaySpecVec, ...
    idata.inv1Dqt.smoothMulti.z(1)/2 + ...
    [-idata.inv1Dqt.smoothMulti.z(1)/2 ...
    idata.inv1Dqt.smoothMulti.z], [M;M(end,:)])
axis ij; box on; shading flat;
set(gca,'Xscale','log');
cc=colorbar; set(get(cc,'YLabel'),'String', 'water ...
    content/ m^3/m^3')
xlabel('Decay time T_2^* [s]'); ylabel('Depth [m]');
```

  For other options and manipulations of the solution, refer to the MRSMatlab code named mrsInvQT_plotData.m.

- Smooth-Mono: In this case, the solution is expressed as two vectors containing the water contents (w) and the decay time (T2). In the case of bootstrap inversions, the solution structure has N dimensions, corresponding to the number of bootstrap that was performed plus one for the classical inversion. To display the solutions of the QT inversion on the whole dataset (no bootstrap), you can type:

```
irun = 1;
T2 = idata.inv1Dqt.smoothMono.solution(irun).T2; % The ...
    decay time distribution
W = idata.inv1Dqt.smoothMono.solution(irun).w; % The water ...
    content distribution
figure('Units','Normalized','Position',[0.1 0.1 0.2 0.8]);
subplot(1,5,1:2)
```

```
stairs([T2(1); T2], [0 idata.inv1Dqt.smoothMono.z])
axis ij; set(gca,'Xscale','log'); grid on
set(gca,'Xminorgrid','off');
xlabel('Decay time T_2^* [s]'); ylabel('Depth [m]');
subplot(1,5,4:5)
stairs([W(1); W], [0 idata.inv1Dqt.smoothMono.z])
axis ij; grid on
set(gca,'Xminorgrid','off');
xlabel('Water content [m^3/m^3]'); ylabel('Depth [m]');
```

In order to plot the other runs (bootstrap), you just need to change the irun value. For further details on the manipulation of the solution, please refer to the MRSMatlab code named mrsInvQT_plotData.m.

- Block-Mono: In this last case, the solution is expresses as a set of variables defining the characteristics of each N layer composing the model. It is stored in three vectors:
  - T2: the decay time ($1 \times N$)
  - w: the water content ($1 \times N$)
  - thk: the thickness ($1 \times (N-1)$)

  As in the previous case, multiple solutions may exist in the case of the use of multiple populations for the Genetic Algorithm. Those different solutions are stored in the different dimensions of the solution structure. To display one solution of the Genetic Algorithm, you can type in the Command Window:

```
irun = 1; % Or any other value depending on the index of ...
    the solution required
T2 = [idata.inv1Dqt.blockMono.solution(irun).T2(1) ...
    idata.inv1Dqt.blockMono.solution(irun).T2];
W = [idata.inv1Dqt.blockMono.solution(irun).w(1) ...
    idata.inv1Dqt.blockMono.solution(irun).w];
Depth = [0 ...
    cumsum(idata.inv1Dqt.blockMono.solution(irun).thk) ...
    max(idata.inv1Dqt.blockMono.z)];
figure('Units','Normalized','Position',[0.1 0.1 0.2 0.8]);
subplot(1,5,1:2)
stairs(T2,Depth);
axis ij; set(gca,'Xscale','log'); grid on
set(gca,'Xminorgrid','off');
xlabel('Decay time T_2^* [s]'); ylabel('Depth [m]');
```

```
subplot(1,5,4:5)
stairs(W,Depth);
axis ij; grid on
set(gca,'Xminorgrid','off');
xlabel('Water content [m^3/m^3]'); ylabel('Depth [m]');
xlim([0 0.6]);
```

In order to plot the solution for other populations, you just need to change the irun value. For further details on the manipulation of the solution, please refer to the MRSMatlab code named mrsInvQT_plotData.m.

Abbreviations

Acknowledgements

References

Akca, İrfan, Thomas Günther, Mike Müller-Petke, Ahmet T. Başokur & Ugur Yaramanci. 2014. Joint parameter estimation from magnetic resonance and vertical electric soundings using a multi-objective genetic algorithm: Multi-objective VES-MRS parameter estimation. en. *Geophysical Prospecting* 62(2). 364–376. http://doi.wiley.com/10.1111/1365-2478.12082, accessed 2018-10-03. DOI:10.1111/1365-2478.12082

Aster, R., B. Borchers & C. Thurber. 2013. *Parameters estimation and inverse problems*. 2nd. Academic Press.

Behroozmand, Ahmad A., Kristina Keating & Esben Auken. 2015. A review of the principles and applications of the nmr technique for near-surface characterization. *Surveys in Geophysics* 36(1). 27–85. https://doi.org/10.1007/s10712-014-9304-0. DOI:10.1007/s10712-014-9304-0

Behroozmand, Esben Auken, Gianluca Fiandaca & Simon Rejkjaer. 2016. Increasing the resolution and the signal-to-noise ratio of magnetic resonance sounding data using a central loop configuration. *Geophysical Supplements to the Monthly Notices of the Royal Astronomical Society* 205(1). 243–256.

Braun, Martina & Ugur Yaramanci. 2008. Inversion of resistivity in Magnetic Resonance Sounding. *Journal of Applied Geophysics* 66(3–4). 151–164. http://www.sciencedirect.com/science/article/B6VFC-4RKMJ07-1/2/43b5fd8f8dbbdfb50e356fb7f18c98ae DOI:10.1016/j.jappgeo.2007.12.004

Costabel, Stephan & Mike Müller-Petke. 2014. Despiking of Magnetic Resonance Signals in time and wavelet domain. *Near Surface Geophysics* 12. 185–197.

Hertrich, Marian. 2008. Imaging of groundwater with nuclear magnetic resonance. en. *Progress in Nuclear Magnetic Resonance Spectroscopy* 53(4). 227–248. http://linkinghub.elsevier.com/retrieve/pii/S0079656508000162, accessed 2018-05-19. DOI:10.1016/j.pnmrs.2008.01.002

Larsen, Jakob Juul, Esben Dalgaard & Esben Auken. 2014. Noise cancelling of MRS signals combining model-based removal of powerline harmonics and multichannel Wiener filtering. *Geophysical Journal International* 196(2). 828–836.

Müller-Petke, Mike, Martina Braun, Marian Hertrich, Stephan Costabel & Jan Walbrecker. 2016. Mrsmatlab—a software tool for processing, modeling, and inversion of magnetic resonance sounding datamrsmatlab. *Geophysics* 81(4). WB9–WB21.

Müller-Petke, Mike & Stephan Costabel. 2014. Comparison and optimal parameter setting of reference-based harmonic noise cancellation in time and frequency domain for surface-NMR. *Near Surface Geophysics* 12. 199–210.

Müller-Petke, Mike & Ugur Yaramanci. 2010. QT inversion — Comprehensive use of the complete surface NMR data set. *Geophysics* 75(4). WA199–WA209. http://library.seg.org/doi/10.1190/1.3471523, accessed 2018-04-03. DOI:10.1190/1.3471523

Trushkin Shushakov, Legchenko. 1994. The potential of a noise-reducing antenna for surface nmr groundwater surveys in the earth's magnetic field. *Geophysical Prospecting* 42(8). 855–862.

Wang, Qi, Chuandong Jiang & Mike Müller-Petke. 2018. An alternative approach to handling co-frequency harmonics in surface nuclear magnetic resonance data. *Geophysical Journal International* 215(3). 1962–1973.