MR-Encephalography (MREG) Pulseq Manual

Jakob Assländer, Pierre LeVan, Bruno Riemenschneider,
Patrick Hucker, Antonia Barghoorn
University Hospital Freiburg, Germany

17.12.2024

1 General Remarks

1.1 Preface

The MREG pulse sequence has been developed in Freiburg over the past 20 years. For the last decade, it has been utilized in its current form, as described in the following publication:

> Assländer J, Zahneisen B, Hugger T, Reisert M, Lee HL, LeVan P, Hennig J, 2013. Single shot whole brain imaging using spherical stack of spirals trajectories. *Neuroimage.* 73:59-70. doi: 10.1016/j.neuroimage.2013.01.065.

This toolbox provides a Pulseq implementation of the standard Siemens sequence. Pulseq is an open framework that facilitates the design, sharing, and execution of MRI sequences across various platforms. For detailed information about the Pulseq framework, refer to the official tutorials available at:

> `https://github.com/pulseq/tutorials`

The sequence works on 1.5 T, 7 T, and 9.4 T systems. However, this particular trajectory is designed for 3 T systems with the 64-channel head coil. Adjustments may be needed for different setups to account for off-resonance effects (higher fields) or reduced encoding power (e.g., at 1.5 T).
Additionally, the trajectory and, in particular, the spoiler gradients may induce peripheral nerve stimulation (PNS) depending on the gradient specifications of the scanner being used. They might need to be adjusted accordingly.
Please note that this initial Pulseq implementation does not currently support tracking physiological signals (e.g., ECG and respiration) using the PMU unit or performing trajectory calibration measurements. FOV positioning is now possible, but only translational, not rotational.

1.2 Disclaimer

The sequence and associated documentation are provided without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose, and noninfringement.

1.3 Revision History

- V 1.0 — 16.12.2024-initial Pulseq release
- V 1.1—- 04.06.2025- FOV positioning now available

2 Method Principles and Implementation

2.1 MR-Encephalography (MREG)

Parallel imaging can greatly decrease MRI acquisition times by exploiting the variable sensitivities of multiple receiver coils rather than time-consuming gradient usage for spatial encoding. Indeed, a first implementation of MREG relied exclusively on coil sensitivities to perform imaging with no gradients at all, resulting in extreme acquisition speeds, but poor spatial resolution [3].
To maintain spatial resolution, initial practical approaches to parallel imaging have consisted of regular k-space undersampling in the phase encode dimension [7],[6],[2]. The achievable undersampling, and thus acceleration, factor is limited by g-factor considerations; coils with high encoding power, and thus low g-factors, tend to consist of a large number of small coil elements tightly and evenly distributed on the surface of the imaged object [9]. Such coils also facilitate parallel imaging in 3D by enabling additional undersampling in a second phase encode dimension [8].
In addition to parallel imaging, further acceleration is possible by using optimized k-space trajectories. MR data are traditionally acquired with Cartesian k-space sampling and regular undersampling patterns, which lead to a computationally simple image reconstruction, however irregular non-Cartesian trajectories can be used for example to perform variable-density sampling, which notably allows the undersampling of k-space areas with sparse information content [4]. Additionally, more efficient gradient usage can be achieved with non-Cartesian trajectories such as spirals [5].
The aim of the MREG sequence is to provide a framework for 3D dynamic imaging using arbitrary k-space trajectories. Notably, a provided 3D variable-density stack-of-spirals trajectory combined with parallel imaging enables single-shot whole-brain acquisitions with 3mm isotropic voxel size and 82ms TR [1]. Nevertheless, users are free to define and use other k-space trajectories according to their needs.

2.2 Implementation

The MREG sequence operates as a gradient echo sequence with stack of spirals readout. Unlike the Siemens implementation, this trajectory is directly converted into an arbitrary gradient waveform and does not require an additional gradient file to be stored on the scanner. However, as of December 16, 2024, the trajectory remains essential for the reconstruction process. Consequently, it is generated and exported by the `MREG_github.m` script.

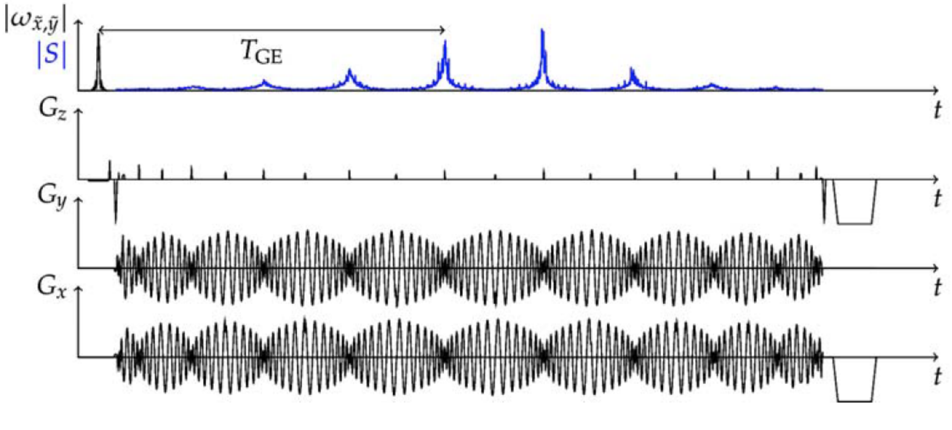

Figure 1: Schematic drawing of the sequence kernel (one TR period) for a stack of spirals trajectory.

The TE displayed in the user interface represents the time from the excitation pulse center to the start of the trajectory. The actual echo time includes the intrinsic time for the trajectory to reach k-space center. In the uploaded MREG.seq file, a large gradient spoiler (X.Zhao, MRM 44:758–765 (2000)) is played out after each read-out to suppress temporal variation in voxel time series due to B0 variations. The amplitude

of the spoiler can be changed in the `MREG_github.m` file for desired visibility of flow artifacts.
The optimal algorithm for image reconstruction is highly dependent on the used k-space trajectory, so that an online ICE reconstruction cannot be provided. Raw data must therefore be exported to an external computer for offline reconstruction. A MATLAB package for MREG image reconstruction is available for this purpose. This package implements tailored algorithms for iterative regularized reconstruction, which is highly computationally intensive, but allows for general, arbitrary trajectories.

3 Sequence and Protocol

3.1 Getting all necessary parts

For using MREG with Pulseq you will need

- This manual
- Pulseq interpreter for your scanner
- Reference protocol as PDF.
- The MREG.seq file for your MR gradient system
- The reconstruction toolbox.

3.2 Manual Installation

The Pulseq sequence interpreter needs to be installed based on the used scanner.

3.3 Sequence Special Card

Sequence parameters such as FOV, TE, TR etc. are set in the `MREG_github.m` file and are NOT set in the protocol at the scanner, as with other Pulseq sequences. In the protocol special card, the following parameters must be set:

- The compiled MREG.seq file needs to be selected as the Pulseq file
- FOV positioning should be enabled
- libBalance should be disabled
- the number of runs multiplied with the TR corresponds to the total scan time
- Delay after and before run(s) should be zero
- Gradient scaling must be 100%

3.4 FOV positioning

Slab excitation is performed by means of a thick 2d-slice selective sinc-pulse in the z-direction (main magnetic field). For 3D imaging, care must be taken when setting the slice (slab) thickness to prevent fold-over artifacts in the z-direction. The relevant geometries are shown in Figure 2.

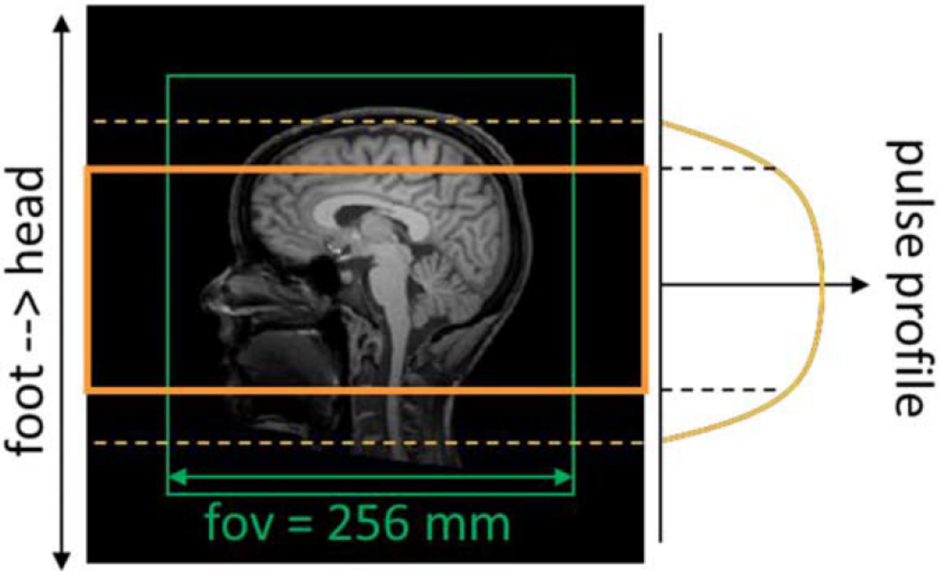

Figure 2: nominal field-of-view, slab excitation and pulse profile.

In this example, the orange rectangle represents the nominal slice thickness selected in the task card. However, the actual excitation also includes contributions from spins within the dashed orange lines due to the imperfect profile of the excitation pulse.

Slice Position and Orientation: Please copy the slice position and orientation from the reference scan in order to make sure that the slices are properly aligned for reconstruction.

3.5 Gradient File Format

In the current implementation the trajectory needs to be exported in a specific gradient file format for image reconstruction. A gradient file is a simple text file with a header and gradient amplitudes for all three gradient axes (read, phase, and slice), stored as N lines with three columns. The header stores the following information:

- `Number_of_Samples`: Corresponds to the total number of lines in the gradient file. The last line read in is defined by this number.
- `Maximum_Gradient_Amplitude_[mT/m]`: Reference amplitude in physical units (mT/m). The gradient samples are stored in the range $[-1, 1]$ and are relative to the reference amplitude. The actual gradient strength (in physical units) is calculated by multiplying the sample value by the reference amplitude.
- `Number_of_Elements`: For multi-shot imaging, this value is greater than 1 and describes the number of different trajectories or trajectory elements (e.g., interleaved k-space coverage) stored within the file.
- `Element_length`: Redundant information. Equal to `Number_of_Samples` divided by `Number_of_Elements`.
- `Field_Of_View_[mm]`: The nominal field of view in millimeters. This value is displayed in the sequence interface.
- `Base_Resolution`: Nominal resolution displayed as the base resolution in the sequence interface.
- `TE_[μs]`: Effective echo time of the trajectory, measured from the start of the trajectory. The actual echo time during the sequence is stored as `TE[1]` in the protocol, where:

$$\texttt{TE[1]} = \texttt{TE[0]} + \texttt{TE\_[}\mu\texttt{s]}$$

  Here, `TE[0]` is the delay specified in the user interface.

- `Dwell_[ns]`: Dwell time for the ADC, stored in nanoseconds.
- `Indexn_start`: Stored in the raw data header, used to predefine ADC samples that should be omitted during reconstruction (e.g., ramp sampling). Different start values can be stored for each trajectory element.
- `Indexn_end`: Similar to `Indexn_start`, used to define the end of the samples.

Below is an example of a gradient file with one trajectory element:

```
##GradientFileHeader##
Number_of_Samples 7454
Maximum_Gradient_Amplitude_[mT/m] 22.8103
Number_of_Elements 1
Element_length 7454
Field_Of_View_[mm] 192 192 192
Base_Resolution 64
TE_[micros] 34780
Dwell_[ns] 5000
##EndOfHeader##
index1_start 72
index1_end 7402
index2_start 0
index2_end 0
...
index16_start 0
index16_end 0
##EndOfIndices##
0.0000 0.0000 0.0000
0.0000 0.0000 0.0000
...
```

Figure 3: Example gradient file with one trajectory element.

3.6 Reference Scan

To perform MREG such as described in (Assländer et al., 2013), that is for 3D singleshot dynamic acquisitions with parallel imaging, a fully encoded reference measurement is required. It is used to estimate the coil sensitivity profiles of the head coil array and provides a map of static off-resonances that can be used for off-resonance corrected reconstruction. It may also be used as an anatomical image (with moderate spatial resolution) of the subject. While there are several ways to fulfill those requirements, we suggest using a standard gradient echo sequence (gre) in 2d multi-slice mode with double echo acquisition.

The imaging geometry should match the nominal FOV specified in the gradient file and should be aligned with the MREG FOV (right click on the protocol allows you to copy the slice position). The easiest way to achieve this is to acquire N slices (slice thickness = nominal voxel size of MREG measurement) with N identical to the base resolution specified in the gradient file. You can increase the resolution of the reference scan for overlaying purposes if you want to. Double-echo acquisition means that a second echo is generated by reversing the readout gradient. The selection of monopolar readout gradients (both readout gradients have identical polarity) avoids phase shifts between both acquired lines in k-space (the same phase shifts lead to the well-known N/2-ghosts in EPI).

In order to avoid additional phase offsets due to the chemical shift of fat, the echo times must be an integer multiple of the fat-water-resonance offset (340Hz at 3T, corresponds to 2.3ms). Then both fat and water spins are in-phase, and the measured phase difference of a voxel is only caused by macroscopic field inhomogeneities. The sequence tooltip usually indicates possible appropriate timings for in-phase and out-of-phase fat and water spins. The long TR due to the multi-slice mode and a flip angle according to the

Ernst angle (or less) ensures that spin history effects can be neglected. Suggested settings for N = 64 at 3T are:

- TR = 1000 ms
- TE1 = 2.3 ms
- TE2 = 4.6 ms
- Flip angle: $\alpha = 50°$
- 64 slices, slice thickness = 3 mm

For image reconstruction, the available MATLAB reconstruction package requires both reference and MREG data to be exported in raw data format.

4 Known Issues

- Trajectory calibration is not supported.
- Reconstruction using the sliding-window technique is not available.
- Multi-shot imaging with more than one trajectory segment is not supported.

References

[1] J. Assländer et al. "Single shot whole brain imaging using spherical stack of spirals trajectories". In: *Neuroimage* 73 (2013), pp. 59–70. DOI: 10.1016/j.neuroimage.2013.01.065.

[2] M.A. Griswold et al. "Generalized autocalibrating partially parallel acquisitions (GRAPPA)". In: *Magn Reson Med* 47.6 (2002), pp. 1202–1210.

[3] J. Hennig, K. Zhong, and O. Speck. "MR-Encephalography: Fast multi-channel monitoring of brain physiology with magnetic resonance". In: *Neuroimage* 34.1 (2007), pp. 212–219.

[4] M. Lustig, D. Donoho, and J.M. Pauly. "Sparse MRI: The application of compressed sensing for rapid MR imaging". In: *Magn Reson Med* 58.6 (2007), pp. 1182–1195.

[5] D.C. Noll et al. "Spiral K-space MR imaging of cortical activation". In: *J Magn Reson Imaging* 5.1 (1995), pp. 49–56.

[6] K.P. Pruessmann et al. "SENSE: sensitivity encoding for fast MRI". In: *Magn Reson Med* 42.5 (1999), pp. 952–962.

[7] D.K. Sodickson and W.J. Manning. "Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays". In: *Magn Reson Med* 38.4 (1997), pp. 591–603.

[8] M. Weiger, K.P. Pruessmann, and P. Boesiger. "2D SENSE for faster 3D MRI". In: *MAGMA* 14.1 (2002), pp. 10–19.

[9] G.C. Wiggins et al. "32-channel 3 Tesla receive-only phased-array head coil with soccer-ball element geometry". In: *Magn Reson Med* 56.1 (2006), pp. 216–223.