



Original software publication

MUPen2DTool: A new Matlab Tool for 2D Nuclear Magnetic Resonance relaxation data inversion



Villiam Bortolotti^a, Leonardo Brizi^b, Anastasiia Nagmutdinova^a, Fabiana Zama^{c,*}, Germana Landi^c

^a Department of Civil, Chemical, Environmental, and Materials Engineering, University of Bologna, Italy

^b Department of Physics and Astronomy, University of Bologna, Italy

^c Department of Mathematics, University of Bologna, Italy

ARTICLE INFO

Article history:

Received 21 March 2022

Received in revised form 12 July 2022

Accepted 15 October 2022

Keywords:

2D NMR data inversion

Inversion of exponential Laplace-type kernels

Multiple penalty regularization

Uniform penalty principle

Adaptive regularization

ABSTRACT

A great variety of applications requires to process two-dimensional NMR (2DNMR) data to obtain information about the materials properties. In order to face the increasing request for software to easily process 2DNMR data, in (Bortolotti et al. (2019) [1]), the authors released Upen2dTool, an open source MATLAB software tool implementing nonnegatively constrained uniform penalty locally adapted L_2 norm-based regularization for 2DNMR data inversion. This paper presents MUPen2DTool a new open-source MATLAB software tool implementing an unconstrained multipenalty regularization method based on L_2 and L_1 norms. The new software MUPen2DTool outperforms Upen2dTool since the implemented uniform multipenalty method allows to compute very accurate 2DNMR data inversion at reduced computational cost. By means of MUPen2DTool, the user can choose among several types of NMR experiments, and the free software provides codes that can be used and extended easily. Furthermore, a MATLAB interface makes it easier to include users own data. The practical use is demonstrated in the reported examples of both synthetic and real NMR data.

© 2022 Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Code metadata

Current code version	v1.5
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-22-00071
Code Ocean compute capsule	https://codeocean.com/capsule/7539434/tree
Legal Code License	GPL v.3
Code versioning system used	none
Software code languages, tools, and services used	Matlab product
Compilation requirements, operating environments & dependencies	
Link to developer manual	https://site.unibo.it/softwareedicam/en/mupen2d
Support email for questions	fabiana.zama@unibo.it

Software metadata

Current software version	v1.5
Permanent link to executables of this version	https://site.unibo.it/softwareedicam/en/mupen2d
Legal Software License	GPL v.3
Computing platforms/Operating Systems	Linux, Microsoft Windows
Installation requirements & dependencies	Matlab
If available, link to user manual - if formally published include a reference to the publication in the reference list	https://site.unibo.it/softwareedicam/en/mupen2d
Support email for questions	villiam.bortolotti@unibo.it

1. Motivation and significance

Nuclear Magnetic Resonance relaxometry of ^1H nuclei (^1H NMR) can give crucial information about the properties of many

* Corresponding author.

E-mail address: fabiana.zama@unibo.it (Fabiana Zama).

materials, ranging from cement [2] to biological tissues [3]. So, for example in case of rock porous media, ^1H NMR permits the accurate estimate of important petrophysical parameters, such as porosity, saturation and permeability. Moreover, borehole ^1H NMR is extensively used in oil and gas reservoir characterization, and recent technological advances have led to tools suitable for environmental applications (see details in [4]). Recently, two-dimensional NMR (2DNMR) techniques are gaining increasing importance in analyzing different types of porous media [5]. Moreover, new computationally intensive applications, such as multidimensional logging and general 3DNMR data inversion [6], require efficient methods for the inversion of 2DNMR data. There is an increasing request for software that can be easily applied to process 2DNMR data to compute 2D parameter distribution (NMR maps) for all these reasons.

Usually, the NMR parameters investigated are relaxation times (longitudinal T_1 and, or transverse T_2) and self-diffusion coefficient (D) as they are sensitive to the local physical environment and can also provide some chemical information. There may be a range of the NMR parameter values that characterize a given system. Therefore, to interpret NMR experiment results correctly and compute the distributions of these parameters, it is necessary to process the experimental data with a robust and accurate inversion procedure.

From a mathematical point of view, the problem of computing the two-dimensional relaxation time distributions from NMR data is a linear ill-posed problem modeled by a Fredholm integral equation with a separable kernel. Moreover, the strong ill-conditioning and the presence of data noise make the inverse problem very challenging. Therefore, a regularization technique reformulates the inversion problem as a non-negatively constrained optimization problem, whose objective function contains a data fitting term and a regularization term.

It has long been known [7] that different inversion models may produce different relaxation maps with substantially equal data fit properties, motivating the considerable amount of inversion methods proposed in the literature. Unfortunately, the software tools and source codes implementing such methods are seldom freely available for testing.

1.1. Previous work

The inversion of NMR data is ill-posed as the stability of the result could not be guaranteed and even small noise could cause changes in the spectrum. The solution to the NMR inversion usually is based on the methods which have been introduced separately by Tikhonov and Twomey. These methodology addresses the use of “smoothing” or regularization parameter and non-negative constraint [8]. If for 1D NMR data an exponential decay fitting method could be used, 2D NMR data is highly attenuated and require the use of multi-dimensional methods. One of the most frequently used methods are Tikhonov regularization, and maximum-entropy methods. Despite various studies on the NMR data inversion problem, there is no universal solution and often algorithms require estimation of redundant regularization parameters. Some of the successfully implemented solutions have been developed, e.g. adapted Butler–Reeds–Dawson (BRD) algorithm [9,10]. In many previously done researches on multi-dimensional data, inversion of the raw data was done with the use of self-developed inverse-Laplace-transform MATLAB program [11,12]. Here we propose a user-friendly software for multi-dimensional NMR data inversion.

We strongly believe that open source software gives a determinant contribution to the progress of knowledge by making it possible for scholars and researchers to compare and improve their achievements. Therefore, starting from 2009, we released *Upwin* [13] and recently *Upen2dTool* [1], an open-source software tool for 2DNMR data inversion. Both *Upwin*

and *Upen2dTool* implement a L_2 -norm locally adapted regularization (in one and two dimensions, respectively), where the automatic computation of the regularization parameters follows the Uniform Penalty (UPEN) principle. This principle has been originally stated in [14,15] for one-dimensional NMR data; successively, in 2016, it has been extended for two-dimensions data [16], and further analyzed and improved in [17,18], including comparisons with a well known approach based on the global L_2 -norm [10].

1.2. New contribution

Even if the method implemented in *Upen2dTool* [1] computes very accurate 2D distributions, its computational cost may be high since it requires the solution of several non-negatively constrained least-squares problems. We believe the new computationally intensive applications, such as multidimensional logging and general 3DNMR data inversion, require inversion methods with reduced computational costs.

For this reason, a new method has been studied and proposed in [19] which represents a substantial change in the inversion strategy, consisting of adding an L_1 penalty term to the locally adapted L_2 term and removing the non-negativity constraint. The new method allows us to substantially improve the computation efficiency of the inversion process and for some kind of 2D data, to obtain even more robust and accurate NMR maps. Please refer to [16,19] for comparisons with models based on separate L_2 and L_1 penalties as in [10,20].

Furthermore, the proposed inversion approach has recently given outstanding results with data related to the transistors trap-state mapping [21], making it appealing to more general application problems modeled by two dimensional exponential kernels.

We now release the Multiple Uniform Penalty 2D Tool (*MUPen2DTool*) open source software implementing the method proposed in [19]. *MUPen2DTool* consists of the source code, software documentation and a user guide which contains an installation guide, a technical description of synthetic NMR tests and input data format. Moreover, *MUPen2DTool* improves the statistical analysis of the inversion process. *MUPen2DTool* comes also with a user friendly GUI (Graphical User Interface) that guides the user in the different steps of the inversion process. Moreover NMR data of several representative examples are available to help the interested user to assess the toolbox efficiency and effectiveness. The GUI makes it easy to handle the inversion parameters and inspect the loaded data, therefore our software can be flexibly used in the analysis of different types of samples. Being *MUPen2DTool* open source and user friendly, we believe that it has a large number of potential users from different application fields.

In this paper we describe the software through a brief overview of the implemented algorithms and a detailed analysis of the 2D distributions computed from the data set enclosed in the software package.

The paper has the following structure. In Section 2 we revise the problem of NMR data inversion and the characteristics of the implemented regularization algorithm. In Section 3 we describe the general structure of *MUPen2DTool* and its key features. Finally, in Section 4, we describe the results of software validation on synthetically generated data as well as acquired NMR data of different relaxometry experiments.

2. The problem of NMR data inversion

In this section, we describe the mathematical model for NMR data inversion and the numerical scheme used by *MUPen2DTool* for its solution.

2.1. The continuous model

In MUPen2DToo1, we consider 2DNMR maps corresponding to $T_1 - T_2$, $T_2 - T_2$ and $D - T_2$ relaxation data; in all these cases, the measured NMR signal is supposed to be related to an underlying distribution function by a Fredholm integral equation of the first kind.

$T_1 - T_2$ case. In a conventional Inversion-Recovery (IR) or Saturation Recovery (SR) experiment detected by a Carr-Purcell-Meiboom-Gill (CPMG) pulse sequence [22], the relaxation data $S(t_1, t_2)$ depending on t_1, t_2 evolution times can be expressed as:

$$S(t_1, t_2) = \iint_0^\infty k_1(t_1, T_1)k_2(t_2, T_2)F(T_1, T_2)dT_1dT_2 + e(t_1, t_2) \quad (1)$$

where $F(T_1, T_2)$ is the unknown distribution of T_1 and T_2 relaxation times and the kernels k_1 and k_2 have the expression

$$k_1(t_1, T_1) = \begin{cases} 1 - 2 \exp(-t_1/T_1), & \text{for IR sequence} \\ 1 - \exp(-t_1/T_1), & \text{for SR sequence} \end{cases} \quad (2)$$

$$k_2(t_2, T_2) = \exp(-t_2/T_2).$$

Here and henceforth, the function $e(\cdot, \cdot)$ represents Gaussian additive noise.

$T_2 - T_2$ case. In a CPMG-CPMG experiment, the measured data $S(t_1, t_2)$ is related to the underlying distribution $F(T_{21}, T_{22})$ by the integral equation

$$S(t_1, t_2) = \iint_0^\infty k_1(t_1, T_{21})k_2(t_2, T_{22})F(T_{21}, T_{22})dT_{21}dT_{22} + e(t_1, t_2) \quad (3)$$

where both kernels k_1 and k_2 refer to transversal relaxation times T_{21}, T_{22} and are defined as

$$k_1(t_1, T_{21}) = \exp(-t_1/T_{21}), \quad k_2(t_2, T_{22}) = \exp(-t_2/T_{22}). \quad (4)$$

Diffusion- T_2 case. In a Stimulated Echo-CPMG experiment, the acquired echo amplitude $S(t_1, t_2)$ can be expressed as

$$S(t_1, t_2) = \iint_0^\infty k_1(t_1, T_2)k_2(t_2, D)F(T_2, D)dT_2dD + e(t_1, t_2) \quad (5)$$

where the kernels k_1 and k_2 are

$$k_1(t_1, T_2) = \exp(-t_1/T_2), \quad k_2(t_2, D) = \exp(-t_2 \cdot D). \quad (6)$$

General exponential Kernel. This kernel is introduced for non-NMR data introduced for more general contexts such as [21]. In this case, $S(t_1, t_2)$ is expressed as

$$S(t_1, t_2) = \iint_0^\infty k_1(t_1, T_1)k_2(t_2, T_2)F(T_1, T_2)dT_1dT_2 + e(t_1, t_2) \quad (7)$$

where the kernels k_1 and k_2 are

$$k_1(t_1, T_1) = 1 - \exp(-t_1/T_1), \quad k_2(t_2, T_2) = 1 - \exp(-t_2/T_2). \quad (8)$$

The objective is to estimate the $T_1 - T_2$, $T_2 - T_2$ or $D - T_2$ maps from the measured data; this inversion is an ill-posed problem, which means that small noise in the data can cause significant changes in the computed 2D distribution.

2.2. The discrete model

The discretization of the linear integral equations (1), (3) and (5) leads to the linear system

$$\mathbf{K}\mathbf{f} + \mathbf{e} = \mathbf{s} \quad (9)$$

where $\mathbf{K} = \mathbf{K}_2 \otimes \mathbf{K}_1$ is the Kronecker product of the discretized kernels $\mathbf{K}_1 \in \mathbb{R}^{M_1 \times N_1}$ and $\mathbf{K}_2 \in \mathbb{R}^{M_2 \times N_2}$, the vector $\mathbf{s} \in \mathbb{R}^M$, $M = M_1 \cdot M_2$, represents the measured noisy signal, $\mathbf{f} \in \mathbb{R}^N$,

$N = N_1 \cdot N_2$, is the vector reordering of the 2D distribution to be computed and $\mathbf{e} \in \mathbb{R}^M$ represents the additive Gaussian noise.

The linear problem (9) is typically ill-conditioned and regularization strategies are necessary to obtain stable discrete distributions. A recent review of regularization techniques used for 2DNMR data inversion can be found in [19].

2.3. The minimization problem

MUPen2DToo1 uses a multipenalty approach based on both $L1$ and $L2$ regularization with locally adapted regularization parameters. In this regularization framework, the NMR data inversion problem is reformulated as the unconstrained minimization problem

$$\min_{\mathbf{f}} \left\{ \|\mathbf{K}\mathbf{f} - \mathbf{s}\|^2 + \sum_{i=1}^N \lambda_i (\mathbf{L}\mathbf{f})_i^2 + \alpha \|\mathbf{f}\|_1 \right\}, \quad (10)$$

where $\|\cdot\|$ denotes the Euclidean norm. The first term of the objective function expresses data consistency in the presence of Gaussian noise while the penalty terms take into account two kind of *a priori* information about the underlying distribution: first, the distribution is known to be a smooth function with some Gaussian-like peaks over flat areas and, second, it is known to be sparse. By using the UPEN principle, the values of the regularization parameters λ_i , $i = 1, \dots, N$ and α can be automatically computed and adapted to the shape of the sought-for distribution [16]. In our previous work [19], we have shown that multipenalty regularization (10), involving $L2$ and $L1$ norm, is able to promote distinct features of the computed distribution, since it produces a good trade-off among data fitting error, sparsity and smoothness of the solution.

2.4. The minimization method

The Fast Iterative Shrinkage and Thresholding (FISTA) method with constant backtracking is used to efficiently compute a solution of (10). We first reformulate problem (10) as the sum of two convex functionals:

$$\min_{\mathbf{f}} \{\Psi_1(\mathbf{f}) + \Psi_2(\mathbf{f})\} \quad (11)$$

where:

$$\Psi_1(\mathbf{f}) = \left\| \begin{pmatrix} \mathbf{K} \\ \sqrt{\Lambda} \end{pmatrix} \mathbf{f} - \begin{pmatrix} \mathbf{s} \\ \mathbf{0} \end{pmatrix} \right\|^2, \quad \Lambda = \text{diag}(\lambda_i) \text{ and } \Psi_2(\mathbf{f}) = \alpha \|\mathbf{f}\|_1.$$

Given a constant stepsize ξ and a starting point $\mathbf{f}^{(0)}$, the FISTA method for (11) generates a sequence of iterates as follows

$$\mathbf{f}^{(j)} = \arg \min_{\mathbf{f}} \left\{ \Psi_2(\mathbf{f}) + \frac{\xi}{2} \left\| \mathbf{f} - \left(\mathbf{y}^{(j)} - \frac{1}{\xi} \nabla(\Psi_1(\mathbf{y}^{(j)})) \right) \right\|^2 \right\} \quad (12)$$

$$t_{j+1} = \frac{1}{2} \left(1 + \sqrt{1 + 4t_j^2} \right) \quad (13)$$

$$\mathbf{y}^{(j+1)} = \mathbf{f}^{(j)} + \frac{(t_j - 1)}{t_{j+1}} (\mathbf{f}^{(j)} - \mathbf{f}^{(j-1)}) \quad (14)$$

with $\mathbf{y}^{(1)} = \mathbf{f}^{(0)}$ and $t_1 = 1$. Since $\Psi_2(\mathbf{f}) = \alpha \|\mathbf{f}\|_1$, the solution of the subproblem (12) can be computed explicitly, element-wise, by means of the soft thresholding operator:

$$\mathbf{f}_i^{(j)} = \text{sign} \left(z_i^{(j)} - \frac{\alpha}{\xi} \right) \max \left(\left| z_i^{(j)} \right| - \frac{\alpha}{\xi}, 0 \right), \quad i = 1, \dots, N \quad (15)$$

where

$$z_i^{(j)} = \mathbf{y}_i^{(j)} - \frac{1}{\xi} \nabla(\Psi_1(\mathbf{y}^{(j)})).$$

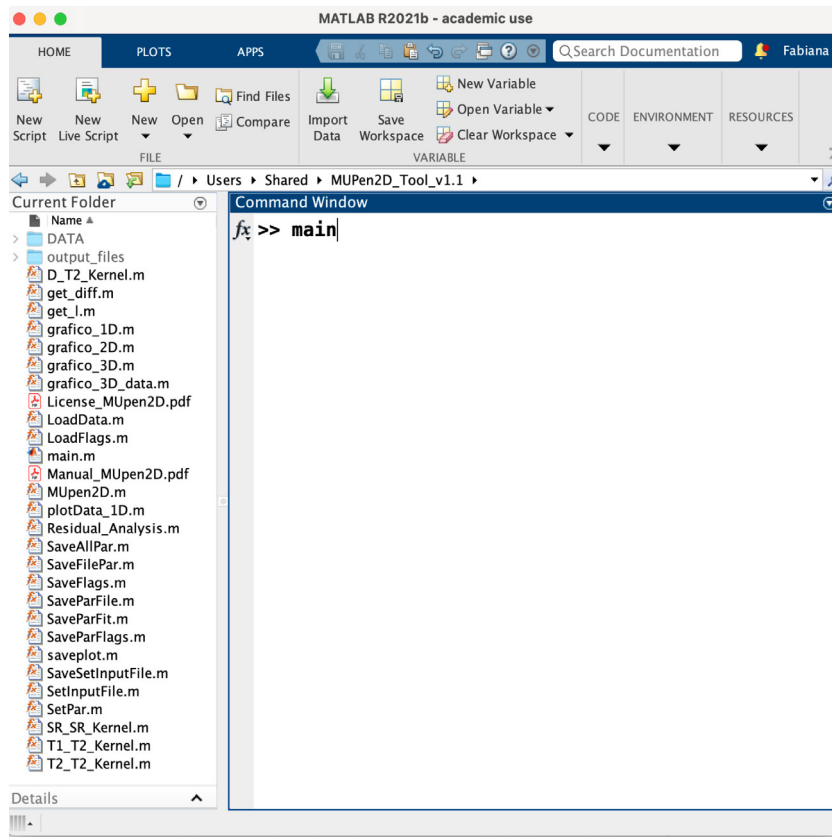


Fig. 1. MATLAB current folder to run MUPen2DTool.

Flags for experiment IR-CPMG	Flags for experiment SSE-CPMG	Flags for V-V experiment
%	%	%
FL_typeKernel =1	FL_typeKernel =3	FL_typeKernel =5
FL_InversionTimeLimits=0	FL_InversionTimeLimits=0	FL_InversionTimeLimits=0
FL_OutputData =1	FL_OutputData =1	FL_OutputData =1
FL_NoContour =15	FL_NoContour =10	FL_NoContour =20
FL_Verbose =0	FL_Verbose =0	FL_Verbose =0
FL_Debug =0	FL_Debug =0	FL_Debug =0
FL_Amp_scale =1	FL_Amp_scale =1	FL_Amp_scale =1
FL_Scale_fact =1	FL_Scale_fact =1	FL_Scale_fact =1
FL_T1T2Filter =0	FL_T1T2Filter =0	FL_T1T2Filter =0
FL_EraseCol =0	FL_EraseCol =0	FL_EraseCol =0
FL_EraseRow =0	FL_EraseRow =0	FL_EraseRow =0
FL_Offset =0	FL_Offset =0	FL_Offset =0
FL_Stat =5	FL_Stat =5	FL_Stat =5
END	END	END

(a)

(b)

(c)

Fig. 2. FileFlag.par files of the three example included in the folder DATA: (a) IR-CPMG kernel (2), (b) D-T2 kernel (6), (c) V-V kernel (8).

Convergence of FISTA has been proved for constant stepsizes ξ equal to a Lipschitz constant of $\nabla(\Psi_1)$ [23]. In MUPen2DTool, we set

$$\xi = \left(\sigma_1^{(1)} \sigma_1^{(2)} \right)^2 + 64 \max_i |\lambda_i| \quad (16)$$

where $\sigma_1^{(1)}$ and $\sigma_1^{(2)}$ are the maximum singular values of the matrices \mathbf{K}_1 and \mathbf{K}_2 , respectively. The choice (16) for the stepsize ξ can be proved to satisfy [19]

$$\lambda_{\max}(\mathbf{K}^T \mathbf{K} + \mathbf{L}^T \mathbf{\Lambda} \mathbf{L}) \leq \xi \quad (17)$$

where $\lambda_{\max}(\cdot)$ denotes the maximum eigenvalue of a matrix. The lower bound (17) ensures FISTA convergence [23].

2.5. Spatially adapted regularization parameters

The choice of the regularization parameters λ_i , $i = 1, \dots, N$, and α is crucial to obtain a meaningful distribution. In [16,19], we developed an automatic selection scheme for spatially adapted regularization parameters by using the UPEN principle. Numerical examples in our works demonstrate the effectiveness of this selection rule.

Given an approximated distribution \mathbf{f} , our automatic selection rule for the regularization parameters can be described as follows:

$$\alpha = \frac{\|\mathbf{K}\mathbf{f} - \mathbf{s}\|^2}{\|\mathbf{f}\|_1}; \quad (18)$$

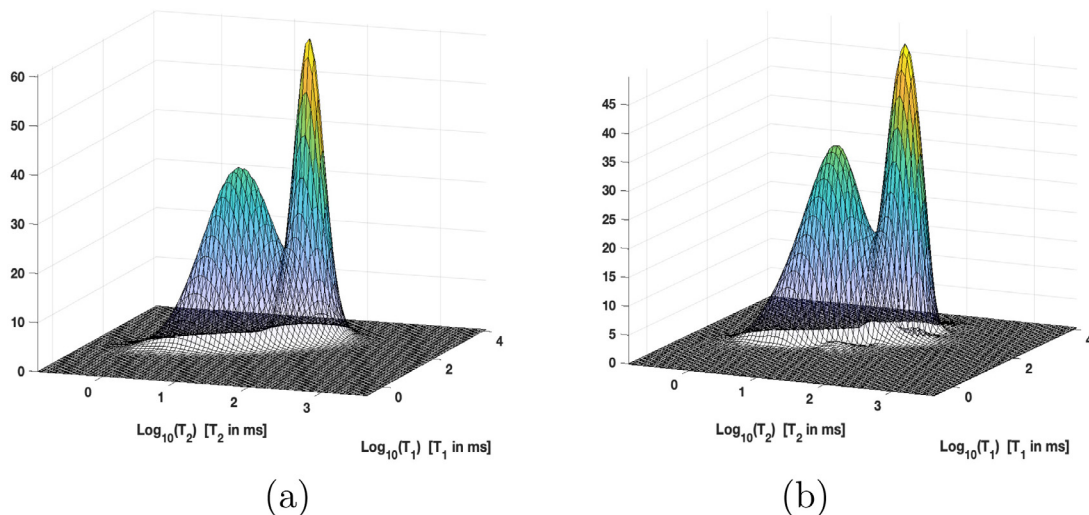


Fig. 3. Relaxation map: (a) reference, (b) computed by MUPen2DTool1.

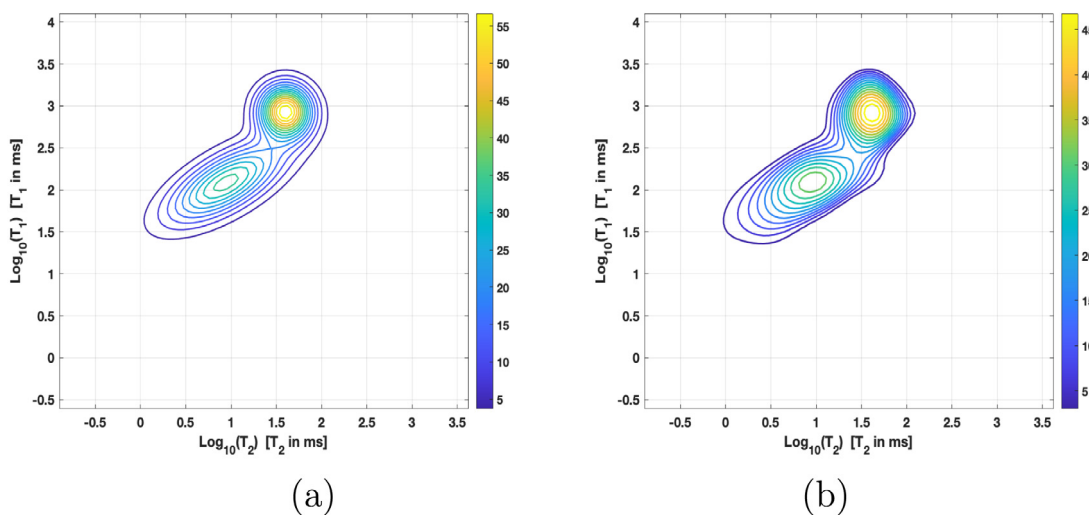


Fig. 4. Contour maps: (a) reference, (b) computed by MUPen2DTool1.

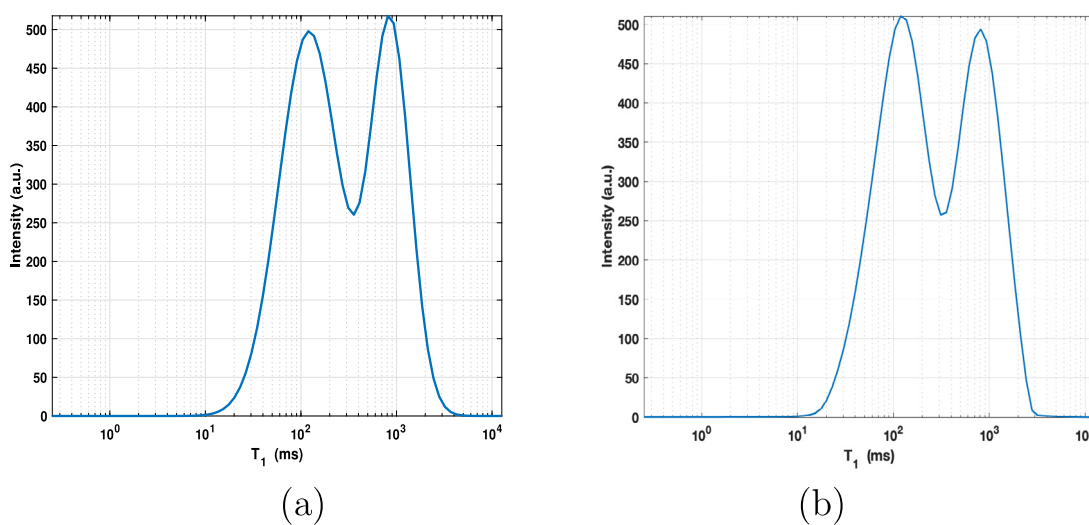


Fig. 5. T_1 one dimension projection: (a) reference, (b) computed by MUPen2DTool1.

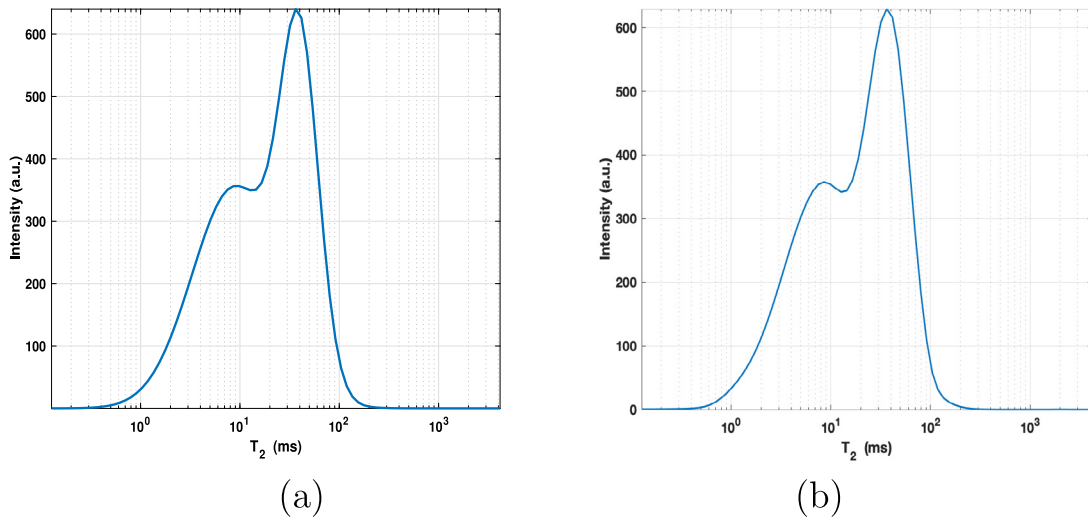


Fig. 6. T_2 one dimension projection: (a) reference, (b) computed by MUPen2DTool.

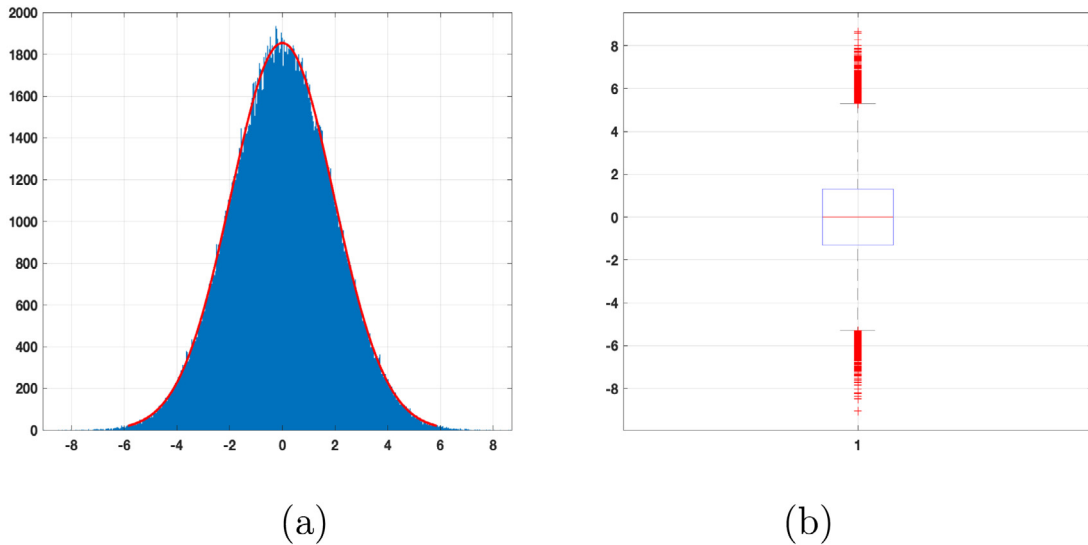


Fig. 7. Post-processing statistics. (a) Histogram of Residual values (blue bars) and Fitted Normal distribution (red line) (b) Box Plot: red horizontal line represents the median value ($-3.5185 \cdot 10^{-3}$); bottom and top edges of the box contain the 25th and 75th percentiles; red '+' marker represents the outliers.

$$\lambda_i = \frac{\|\mathbf{Kf} - \mathbf{s}\|^2}{N \left(\beta_0 + \beta_p \max_{\mu \in I_i} \{\text{vec}(\|\nabla \mathbf{F}\|_\mu)^2\} + \beta_c \max_{\mu \in I_i} \{(\mathbf{L}\mathbf{f})_{\mu 1}^2\} \right)}, \quad i = 1, \dots, N; \quad (19)$$

where $\text{vec}(\cdot)$ denotes the operator that stacks a matrix column-wise to produce a column vector and \mathbf{F} is the 2D distribution map corresponding to \mathbf{f} (i.e., $\mathbf{f} = \text{vec}(\mathbf{F})$). The indices subsets I_i , $i = 1, \dots, N$, are related to the neighborhood of the point i and the β 's are positive parameters whose optimum values can change with the nature of the measured sample. Please refer to [16], for a detailed description of the properties of the β parameters.

2.6. The implemented algorithm

The method implemented in MUPen2DTool computes the solution of a weighted inversion problem based on (10), i.e.:

$$\min_{\mathbf{f}} \left\{ \|\mathbf{Kf} - \mathbf{s}\|^2 + \omega_1 \sum_{i=1}^N \lambda_i (\mathbf{L}\mathbf{f})_i^2 + \omega_2 (\alpha \|\mathbf{f}\|_1) \right\}, \quad \omega_1, \omega_2 \in [0, 1] \quad (20)$$

where the default values of the weights are $\omega_1 = \omega_2 = 1$, as in (10). Depending on the data and problem type, it is possible to introduce further model flexibility by setting ω_i s.t. $\omega_1 + \omega_2 = 1$. This feature is implemented by setting a specific software flag discussed in the subsequent section.

Problem (20) is solved by the FISTA algorithm (equations: (11)–(15)), using the rules (18)–(19) for the automatic computation of the spatially adapted regularization parameters. We report the steps of our method in Algorithm 1. Concerning the starting guess $\mathbf{f}^{(0)}$ we apply a few iterations of the Gradient Projection (GP) method to the following constrained least squares problem, i.e.

$$\mathbf{f}^{(0)} = \arg \min_{\mathbf{f} \geq \ell} \|\mathbf{Kf} - \mathbf{s}\|^2.$$

where $\ell \geq 0$ is mostly used with NMR data, but in general contexts $\ell < 0$ can be found [24].

3. Software description

A MATLAB toolbox is a collection of user-written MATLAB files, with functions and/or classes, aimed at addressing a

Algorithm 1

- 1: Choose $\tau \in (0, 1)$, $\beta_0, \beta_p, \beta_c > 0$ and compute $\sigma_1^{(1)}$ and $\sigma_1^{(2)}$.
- 2: **Initialization**
Set $k = 0$ and choose a starting guess $\mathbf{f}^{(0)}$.
- 3: **repeat**
- 4: **Regularization parameters update**
Set

$$\alpha^{(k)} = \frac{\|\mathbf{K}\mathbf{f}^{(k)} - \mathbf{s}\|^2}{\|\mathbf{f}^{(k)}\|_1};$$

$$\lambda_i^{(k)} = \frac{\|\mathbf{K}\mathbf{f}^{(k)} - \mathbf{s}\|^2}{N \left(\beta_0 + \beta_p \max_{\mu \in I_i} \{ \text{vec}(\|\nabla \mathbf{F}^{(k)}\|_\mu^2) \} + \beta_c \max_{\mu \in I_i} \{ (\mathbf{L}\mathbf{f}^{(k)})_\mu^2 \} \right)},$$

$$i = 1, \dots, N;$$
- 5: **Solution update**
Compute

$$\mathbf{f}^{(k+1)} = \underset{\mathbf{f}}{\text{argmin}} \left\{ \left\| \left(\frac{\mathbf{K}}{\sqrt{\omega_1 \Lambda^{(k)} \mathbf{L}}} \right) \mathbf{f} - \begin{pmatrix} \mathbf{s} \\ \mathbf{0} \end{pmatrix} \right\|^2 + \omega_2 \alpha^{(k)} \|\mathbf{f}\|_1 \right\}$$

by the FISTA method with starting guess $\mathbf{f}^{(k)}$ and constant stepsize $\xi^{(k)} = (\sigma_1^{(1)} \sigma_1^{(2)})^2 + 64 \max_i |\lambda_i^{(k)}|$
- 6: **until** $\|\mathbf{f}^{(k+1)} - \mathbf{f}^{(k)}\| \leq \tau \|\mathbf{f}^{(k)}\|$

specific topic. MUPen2DTool is a collection of functions related to different aspects of the inversion of 2DNMR data. The package, available at <https://doi.org/10.6084/m9.figshare.18197399>, contains all MATLAB scripts and functions to run MUPen2DTool from the MATLAB environment. Alternatively, it is possible to test MUPen2DTool using the MATLAB app GUI_MUPen2D, a dedicated stand alone compiled GUI application available from <https://site.unibo.it/softwareedicam/en/mupen2d>. After downloading MUPen2DTool.zip, extract the content to the folder MUPen2DTool, set the MATLAB current folder as reported in Fig. 1, and run the script main.m in the MATLAB command window.

3.1. Software architecture

The implemented software functions, contained in the MUPen2DTool folder, can be classified as follows:

- Initialization, data load, output data and figures: SetInputFile.m, SetPar.m, LoadData.m, LoadFlags.m, grafico_1D.m, grafico_2D.m, grafico_3D.m.
- Functions that implement the inversion algorithm and post-processing: MUPen2D.m, T1_T2_Kernel.m, T2_T2_Kernel.m, D_T2_Kernel.m, get_diff.m, get_l.m, Residual_Analysis.m.
- Inversion data folder DATA, organized into sub-folders, as detailed below.
- User manual: Manual_Mupen2D.pdf.
- Folder output_files, generated by main, containing data and inversion information, please refer to the software Manual for more details.

Description of DATA folder. The software package includes the following data sub-folders for testing:

- T1-T2_synth_2pks relative to a synthetic $T_1 - T_2$ data numerically generated;
- T1-T2 relative to $T_1 - T_2$ data from a cement sample;

```

=====
MUPen2D Input Parameters
upen_tol=1.000000e-04,
Projected Gradient Tol =1.000000e-04
SVD Threshold =1e-16
Data size= 80 x 80

=====
Number of Inversion channels: horizontal 80, vertical 80
Final Relative Residual Norm =2.5422e-03
Total MUPen2D Iterations = 6
Total FISTA Iterations = 64018
Computation Time = 43.20755 s.

=====
MUPen2D - T2=41.53 T1=814.97 peak=49.90 PercTot=34.06
MUPen2D - (T2= 36.41 41.53 47.38, T1 = 710.55 814.97 934.73)

=====
Statistical Analysis
Normal Distribution : mu = -4.120540e-05, sigma = 1.953138e+00
75th percentile = 1.323289e+00
median = -3.518547e-03
25th percentile = -1.318074e+00
skewness = -3.617721e-04, kurtosis = 2.983085e+00
=====
    
```

Fig. 8. Output file Parameters.txt for the test problem T1-T2_synth_2pks.

- D-T2 relative to $D - T_2$ data from a fresh cement sample.
- V - V relative to synthetic numerically generated data from a known map, using the kernels (8).

Each data folder contains six files, three data files:

- a column ASCII data file with the list of the M time values used to acquire the relaxation curve on the first dimension,
- a column ASCII data file with the list of the N time values used to acquire the relaxation curve on second dimension,
- a 2D $M \times N$ matrix data file with the acquired signal,

which correspond to the files reported in FileSetInput.par, one flag file: FileFlag.par, containing the keywords that define the type of experiment to be processed and one parameter file: Filepar.par, addressing the algorithm and modify the functionality of MUPen2DTool. The detailed description of each keyword can be found in the user manual, here we only outline the main structure. In FileFlag.par we select the type of relaxation data (FL_typeKernel), the left and right extreme limits of the relaxation times (FL_InversionTimeLimits). Fig. 2 reports the FileFlag.par files of the three example included in the folder DATA.

The file FileSetInput.par contains the name of the 2D data file Filenamedata, the names of the files containing the vectors of times in the first (filenameTimeX) and second (filenameTimeY) dimensions and number the relaxation time bins along the first (nx) and second (ny) dimension.

Finally FilePar.par contains the parameters of the inversion algorithm, such as tolerances, maximum number of iterations allowed, the weights in L_2 penalty function (19) ($\beta_0, \beta_p, \beta_c$) and the value par.fista.weight to set the weights ω_1, ω_2 as follows:

```

if par.fista.weight ∈ [0, 1],
{
  ω1 = 1 - par.fista.weight
  ω2 = par.fista.weight
}
else
  ω1 = ω2 = 1
    
```

Usually there is no need to set par.fista.weight ∈ [0, 1] however, depending on the acquired data, the results might be improved.

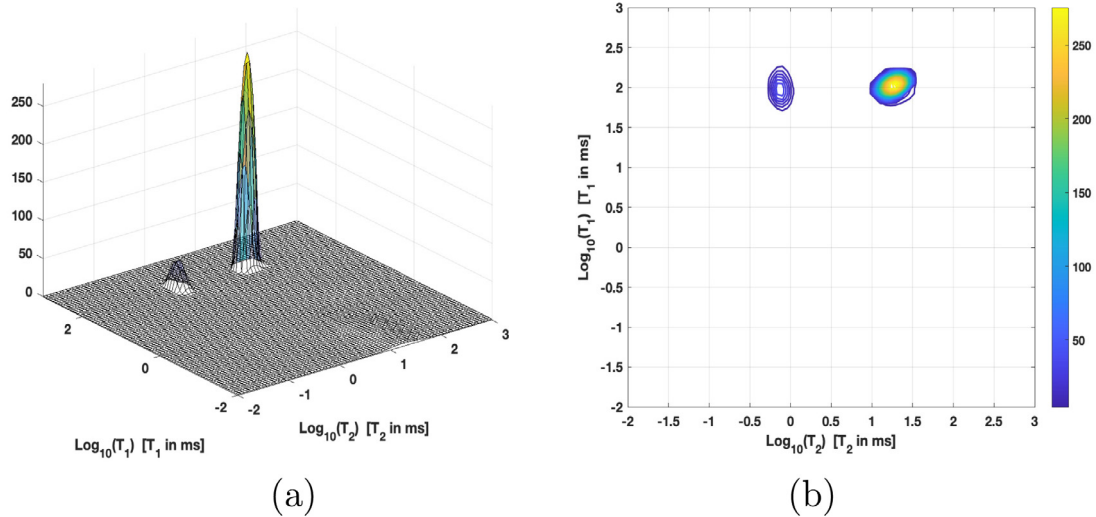


Fig. 9. Maps computed by MUPen2DTool for the $T_1 - T_2$ dataset. (a) surface, (b) contour map.

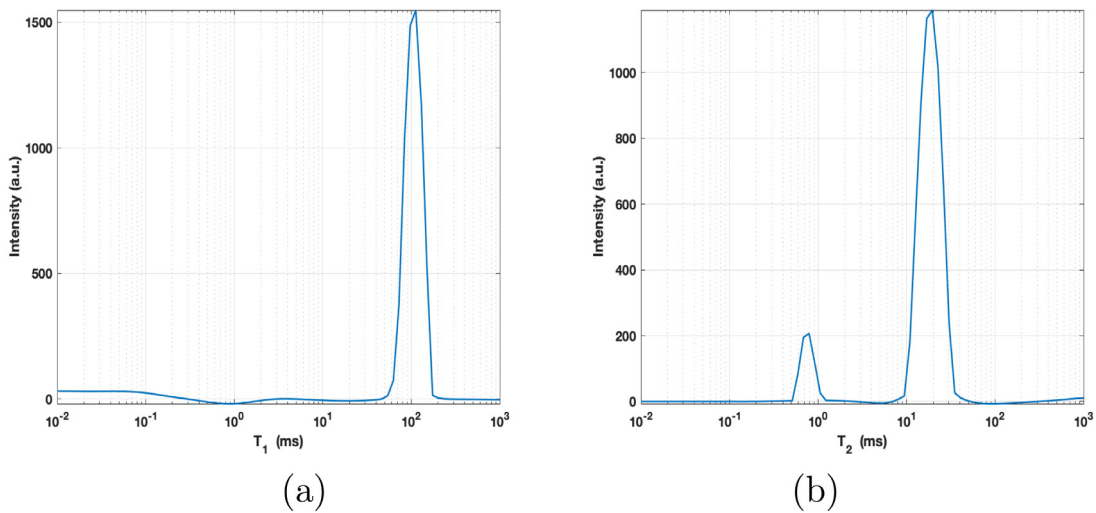


Fig. 10. MUPen2DTool results for the $T_1 - T_2$ dataset. (a) T_1 one dimension projection. (b) T_2 one dimension projection.

```

=====
MUPen2D Input Parameters
  open_tol=1.000000e-04,
  Projected Gradient Tol =1.000000e-04
  SVD Threshold =1e-16
  Data size= 48 x 80
=====
Number of Inversion channels: horizontal 80, vertical 80
Final Relative Residual Norm =2.3211e-02
Total MUPen2D Iterations = 9
Total FISTA Iterations = 10542
Computation Time = 4.10283 s.
=====
MUPen2D - T2=19.55 T1=112.37 peak=280.87 PercTot=85.48
MUPen2D - (T2= 16.90 19.55 22.62, T1 = 97.13 112.37 129.99)
=====
Statistical Analysis
Normal Distribution : mu = -2.033376e-02, sigma = 3.101324e+01
75th percentile = 2.051574e+01
median          = 2.124887e-01
25th percentile = -2.034167e+01
skewness       = -3.587174e-02, kurtosis = 3.369759e+00
=====
    
```

Fig. 11. Output file Parameters.txt for the $T_1 - T_2$ dataset.

4. Illustrative examples

In this section we present a few examples of the results computed by MUPen2DTool on the dataset included in the software package, with the purpose to help users to prepare their own data. The included data folder contains two examples where the data is synthetically computed by a known map ($T_1-T_2_{synth_2pks}$, $V-V_{synth_2pks}$), and two real data examples of $T_1 - T_2$ and $D - T_2$ measurements of fresh Waite Portland Cement (WPC) Cement samples.

The reported results have been obtained by using a PC equipped with a 2.9 GHz Intel microprocessor i7 quad-core and 16 GB RAM.

4.1. $T_1 - T_2$ synthetic numerical data

This example consists of synthetic data generated by the source code main_make_synt.m contained in the folder T1-T2_synth_2pks. To run this test we must select the folder T1-T2_synth_2pks either launching the main.m script in the MATLAB environment or directly the GUI application.

The reference relaxation map $F(T_1, T_2)$, represented in Fig. 3(a), has size 80×80 and presents two peaks at positions

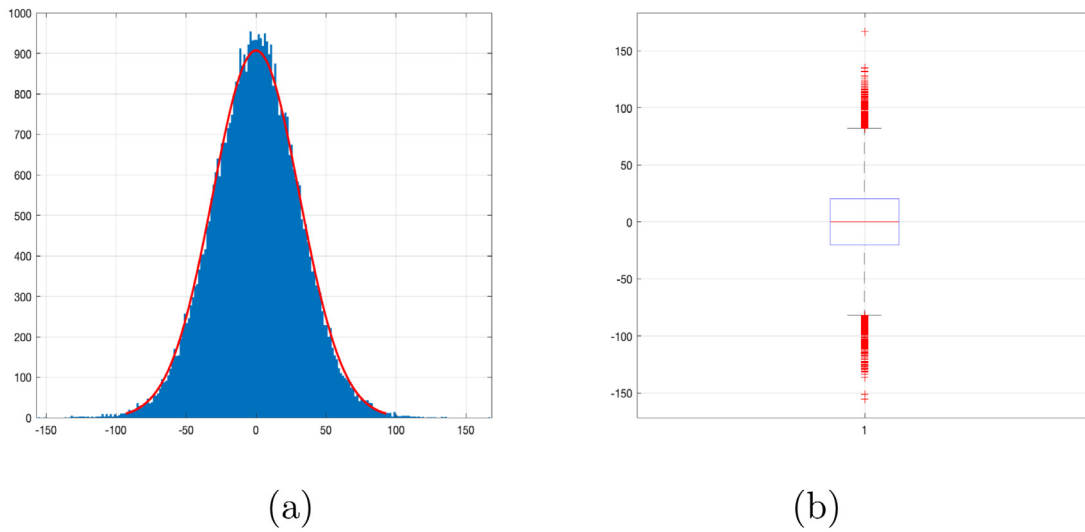


Fig. 12. Post-Processing. (a) Histogram of Residual values (blue bars) and Fitted Normal distribution (red line) (b) Box Plot: red horizontal line represents the median value ($2.123 \cdot 10^{-1}$); bottom and top edges of the box contain the 25th and 75th percentiles; red '+' marker represents the outliers.

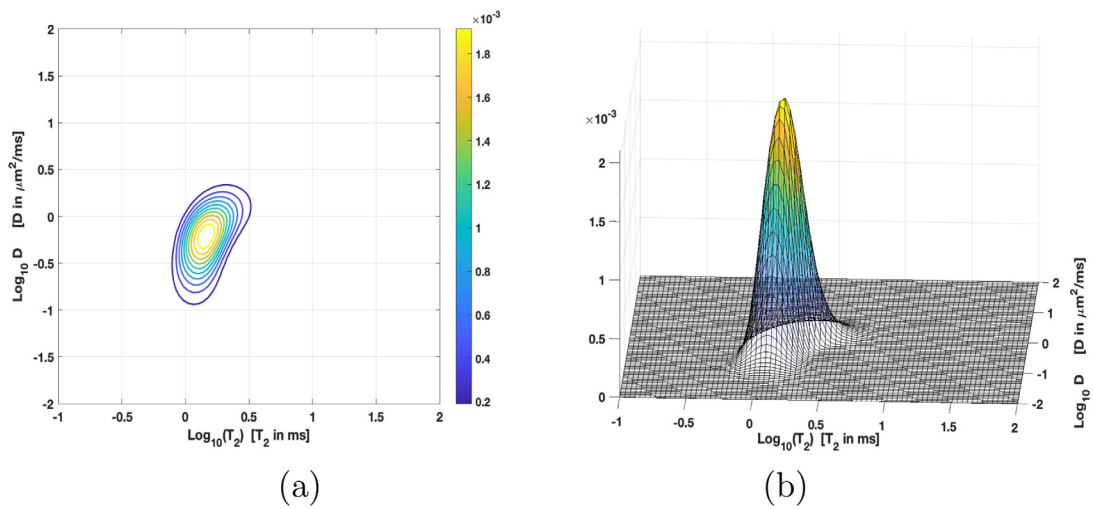


Fig. 13. (a) Contour map and (b) relaxation map computed by MUPen2DTool.

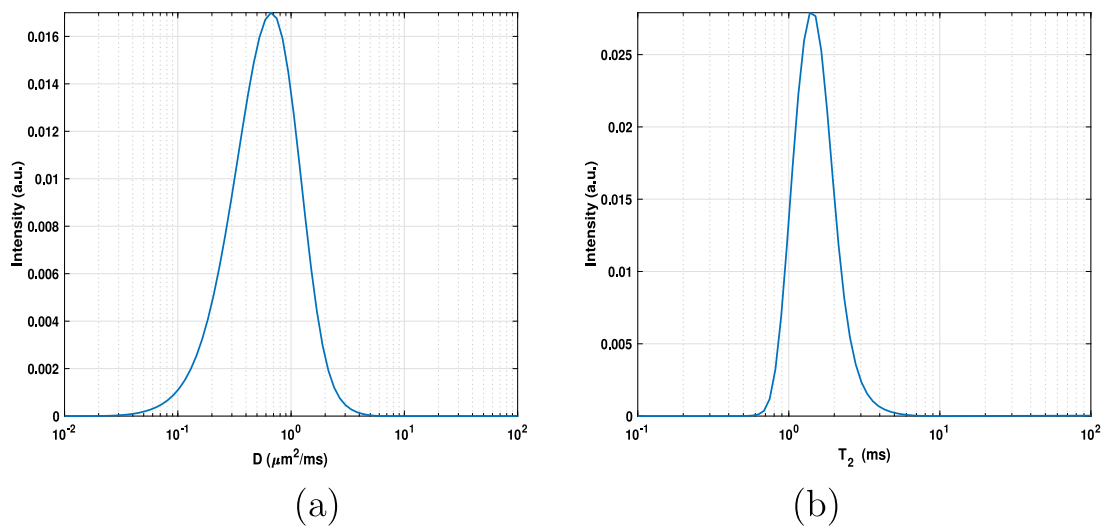


Fig. 14. D one dimension projection (a) and T_2 one dimension projection (b).

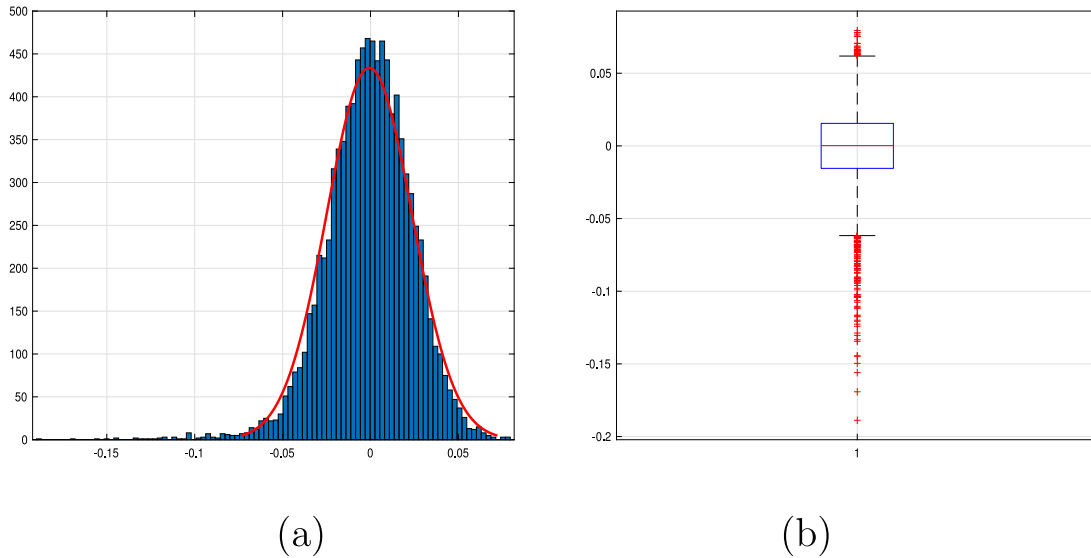


Fig. 15. Post-Processing. (a) Histogram of Residual values (blue bars) and Fitted Normal distribution (red line) (b) Box Plot: red horizontal line represents the median value ($1.7224 \cdot 10^{-4}$); bottom and top edges of the box contain the 25th ($-1.5454 \cdot 10^{-2}$) and 75th ($1.5496 \cdot 10^{-2}$) percentiles; red '+' marker represents the outliers.

```

-----
MUPen2D Input Parameters
  upen_tol=1.000000e-04,
  Projected Gradient Tol =1.000000e-03
  SVD Threshold =1e-16
  Data size= 48 x 80
-----

Number of Inversion channels: horizontal 80, vertical 80
Final Relative Residual Norm =3.6385e-01
Total MUPen2D Iterations = 27
Total FISTA Iterations = 38895
Computation Time = 18.61967 s.
-----

=====
  Statistical Analysis
  Normal Distribution : mu = -6.753610e-04, sigma = 2.426962e-02
  75th percentile = 1.551223e-02
  median          = 1.821731e-04
  25th percentile = -1.545448e-02
  skewness       = -6.253064e-01, kurtosis = 5.404664e+00
=====

```

Fig. 16. Output file Parameters.txt for the test problem D-T2.

($T_1 = 815.0$ ms, $T_2 = 4.533$ ms) and ($T_1 = 119.5$ ms, $T_2 = 8.561$ ms). The reference map is applied to (1) with the IR kernel to obtain the synthetic relaxation data $S(t_1, t_2)$ corresponding to an IRCPMG experiment. The number of IR inversion times is $M_1 = 128$ while the CPMG has $M_2 = 2048$ echoes. Normal Gaussian random noise $\mathbf{e} \in \mathbb{R}^{128 \times 2048}$ of level $\delta \equiv \|\mathbf{e}\| = 10^{-2}$ is added.

The computed relaxation map displayed in Fig. 3(b) shows an accurate representation of the reference map. The detailed discussion of this test in terms of errors and comparison with different solution strategies can be found in [19].

Fig. 4 shows the contour map computed by MUPen2DToo1 and the contour reference map from which we can appreciate the accurate localization of the peaks. The T_1 , T_2 projections are reported in Figs. 5 and 6 for both the MUPen2DToo1 computed map and the reference map. The post-processing step computes some interesting information about the residual $\mathbf{R} = \mathbf{s} - \mathbf{Kf}_{\text{computed}}$.

The histogram of the residual \mathbf{R} , in Fig. 7(a), shows the good agreement of the residual values (blue bars) to the normal distribution (red line) with variance $\sigma = 1.9531$ and mean $\mu = -4.1205 \cdot 10^{-5}$. Moreover, the box-plot in Fig. 7(b) represents the box having the bottom edge given by the 25th percentile (-1.3181) and the top edge given by the 75th percentile (1.3233), containing 260444 out of 262144 points (99.35%). The median value ($-3.5185 \cdot 10^{-3}$) is represented as an horizontal red line. The 1700 outliers are represented by red '+' markers.

The file Parameters.txt in the output sub-folder reports information about input tolerances, data size, algorithm iterations, computation time and statistical analysis (Fig. 8). Information about the lack of symmetry (skewness) and weight of the distribution tails (kurtosis) is contained in the structure out_data, and reported in the file Parameters.txt (Fig. 8) i.e.:

```

out_data.skewness = -3.6177 10-4,
out_data.kurtosis = 2.983.

```

Following [25], \mathbf{R} is considered to be normal since skewness is between -2 and $+2$ and kurtosis is between -7 and $+7$, demonstrating the goodness of the inversion performed.

4.2. $T_1 - T_2$ dataset: cement sample

The data represent a test on fresh cement pastes prepared by filling a 10 mm external diameter glass NMR tube with White Portland Cement mixed with distilled water for 2 min at 1600 r.p.m. (w/c ratio 0.5). The NMR measurements were performed at 25 °C by a homebuilt relaxometer based on a PC-NMR portable NMR console (Stelar, Mede, Italy) and a 0.47 T Jeol electromagnet (Jeol Ltd. Tokyo, Japan). The IR-CPMG 2D data has size 48×1000 and the reconstructed map $F \in \mathbb{R}^{N_1 \times N_2}$ 80×80 ($N_1 \times N_2$). The SVD projection step, with a threshold as the machine epsilon ($\tau = 10^{-16}$), allows a reduction of the data size $2840 = 48 \times 80$. The reconstructed 3D map and 2D contour map are reported in Fig. 9 while the projections onto the T_1 and T_2 axes are reported in Fig. 10. The file Parameters.txt in the output sub-folder reports information about input tolerances, data size, algorithm iterations, computation time and statistical analysis (Fig. 11). The histogram of the residual, represented in Fig. 12(a), shows the good agreement of the residual values (blue bars) to the normal

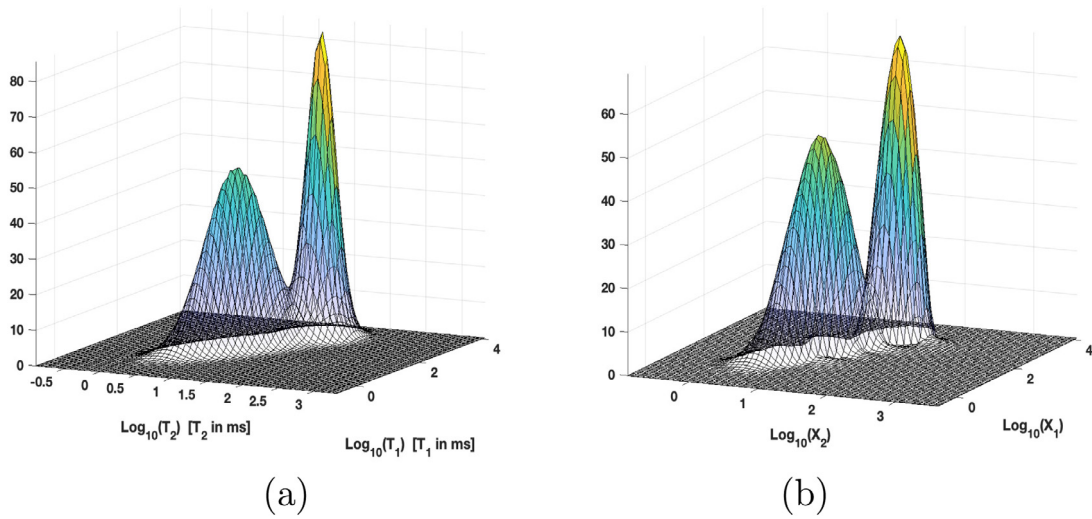


Fig. 17. Reference map (a). MUPen2DToo1computed map (b).

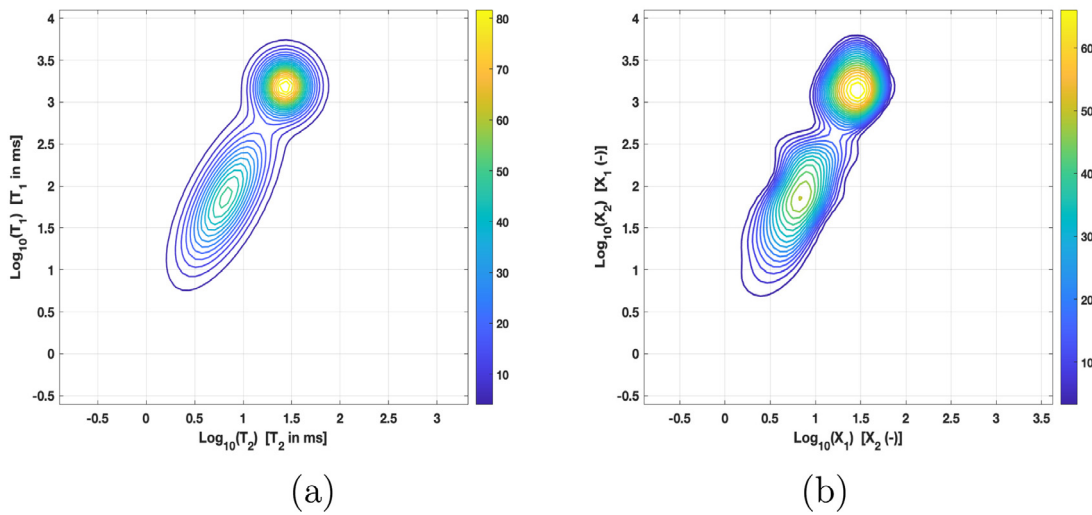


Fig. 18. Contour of reference map (a). Contour of MUPen2DToo1computed map (b).

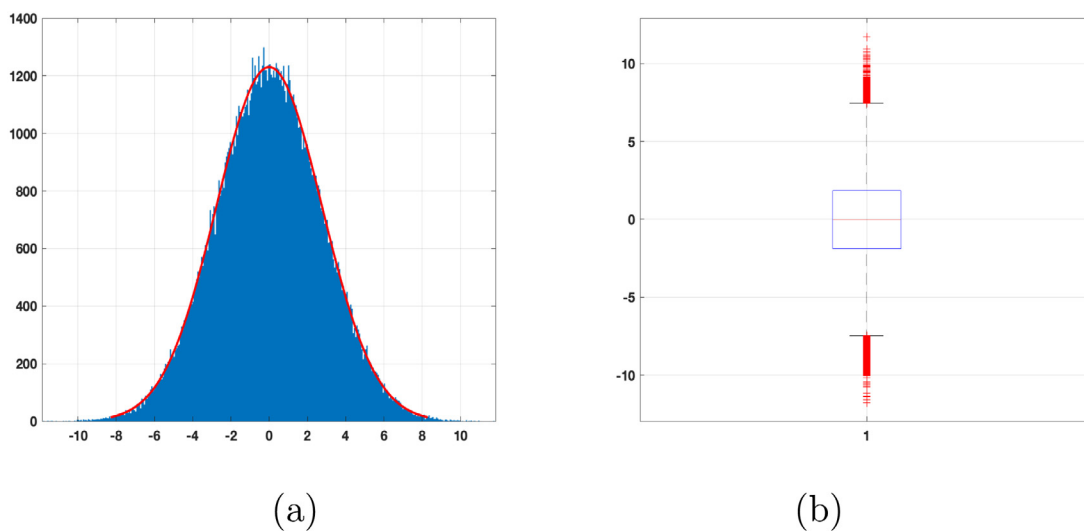


Fig. 19. Post-Processing. (a) Histogram of Residual values (blue bars) and Fitted Normal distribution (red line) (b) Box Plot: red horizontal line represents the median value ($-2.08 \cdot 10^{-3}$); bottom and top edges of the box contain the 25th and 75th percentiles; red '+' marker represents the outliers.

```

MUPen2D Input Parameters
upen_tol=1.000000e-04,
Projected Gradient Tol =1.000000e-04
SVD Threshold =1e-16
Data size= 64 x 64
-----
Number of Inversion channels: horizontal 64, vertical 64
Final Relative Residual Norm =5.3467e-04
Total MUPen2D Iterations = 6
Total FISTA Iterations = 131234
Computation Time = 41.67781 s.
-----
=====
Statistical Analysis
Normal Distribution : mu = -1.626133e-04, sigma = 2.762155e+00
75th percentile = 1.864808e+00
median          = -2.079876e-03
25th percentile = -1.869517e+00
skewness        = -1.183784e-03, kurtosis = 2.982981e+00
=====

```

Fig. 20. Output file Parameters.txt for the test problem V-V_synth_2pks.

distribution (red line) with variance $\sigma = 3.1013 \cdot 10^1$ and mean $\mu = -2.0333 \cdot 10^{-2}$.

4.3. $D - T_2$ dataset: cement sample

The acquired raw data are relative to a fresh WPC sample measured with Stimulated Echo Diffusion Editing sequence SSE-CPMG with the use of a NMR Mouse PM10 set-up (Magritek, NZ) with the field gradient of 14 T/m, size of the sensitive volume $x, y, z [15 \times 15 \times (0.1-0.3)] \text{ mm}^3$, magnetic field in the sensitive volume 0.327 T driven with a KEAll console (Magritek, NZ) and using the following acquisition parameters: CPMG TE= 400 μs , Number of echoes 200, number of scans 200, repetition delay 1.5 s.

Fig. 13 shows the contour and the relaxation map computed by MUPen2DToo1. The projections on the diffusion D and T_2 directions are shown in Fig. 14, where is shown a single T_2 peak at about 10 ms, which probably corresponds to capillary water. With a TE of 400 μs shorter peaks (for example interlayer water) are not detected.

Finally, information about the reconstruction quality can be obtained by the post-processing step. The histogram of the residual distribution in Fig. 15(a) shows a good agreement with the normal distribution with variance $\sigma = 0.024$ and mean $\mu = -6.754 \cdot 10^{-4}$ represented in red line. Information about input tolerances, data size, algorithm iterations, computation time and statistical analysis is contained in the file Parameters.txt (output sub-folder) represented in Fig. 16.

The lack of symmetry (skewness= -0.6265) and weight of the distribution tails (kurtosis= 5.4128) agree with normal data, since skewness is between -2 and $+2$ and kurtosis is between -7 and $+7$ [25].

The box-plot in Fig. 15(b) shows the median ($1.7224 \cdot 10^{-4}$) in the central mark, and the 25th ($-1.5454 \cdot 10^{-2}$) and 75th ($1.5496 \cdot 10^{-2}$) percentiles at the bottom and top edges of the box containing 9400 out of 9600 points (97.9%). The outliers, amounting to 150 points, are plotted individually using red '+' marker symbols. The remaining 50 points lie in the intervals $[-0.06347, -0.0155]$ and $[0.0155, 0.061]$, represented by the

lower and upper whiskers and the box lower and upper boundaries.

4.4. $V - V$ Data

This example consists of synthetic data generated by the source code main_make_synt.m contained in the folder V-V_synth_2pks. To run this test we must select the folder V-V_synth_2pks by launching the main.m script in the MATLAB environment. The reference map $F(T_1, T_2)$, represented in Fig. 17(a), has size 64×64 . The reference map is applied to (7) with kernels (8) to obtain the synthetic data S of size 128×1024 . Normal Gaussian random noise $\mathbf{e} \in \mathbb{R}^{128 \times 1024}$ of level $\delta \equiv \|\mathbf{e}\| = 10^{-2}$ is added.

The good accuracy of the computed map can be appreciated in Figs. 17, 18(b) compared to the reference maps reported in column (a) of the same figures.

The histogram of the residual, represented in Fig. 19(a), shows the good agreement of the residual values (blue bars) to the normal distribution (red line) with variance $\sigma = 2.76224$ and mean $\mu = -1.506 \cdot 10^{-4}$. The file Parameters.txt in the output sub-folder reports information about input tolerances, data size, algorithm iterations, computation time and statistical analysis (Fig. 20). Information about the lack of symmetry (skewness) and weight of the distribution tails (kurtosis) is contained in the structure out_data, and reported in the file Parameters.txt (Fig. 20) i.e.:

```

out_data.skewness = -1.184 10-3,
out_data.kurtosis = 2.983.

```

5. Impact

The impact of MUPen2DToo1 can be summarized as follows.

- Possibility to study a large variety of existing and possibly new materials by means of NMR-relaxometry.
- Inclusion of different kernel types extending the application ranges to non-NMR samples. One first example of such an extension is provided in [21].
- Improvement of the evaluation of the modeling results by descriptive statistics tools.

Concerning the industrial impact, a collaboration is currently in progress with one of the most important companies in the field of NMR-relaxometry, such as Stellar s.r.l.

6. Conclusions

This algorithm paper describes the MUPen2DToo1 open source software implementing the method proposed in [19] for the inversion of 2D NMR data. Several representative examples are presented in detail to help the interested users to include their own data.

We believe that MUPen2DToo1 can be usefully applied in a wide application range not necessarily connected to NMR-relaxometry experiments.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data available at <https://site.unibo.it/softwareedicam/en/mupen2d>

Acknowledgment

This work was partially supported by the Istituto Nazionale di Alta Matematica, Gruppo Nazionale per il Calcolo Scientifico (INdAM-GNCS), Italy.

References

- [1] Bortolotti V, Brizi L, Fantazzini P, Landi G, Zama F. Upen2Dtool: A uniform penalty matlab tool for inversion of 2D NMR relaxation data. *SoftwareX* 2019;10:100302. <http://dx.doi.org/10.1016/j.softx.2019.100302>.
- [2] Nagmutdinova A, Brizi L, Fantazzini P, Bortolotti V. Investigation of the first sorption cycle of white portland cement by ^1H NMR. *Appl Magn Reson* 2021;52:1767–85.
- [3] Fantazzini P, Galassi F, Bortolotti V, Brown R, Vittur F. The search for negative amplitude components in quasi-continuous distributions of relaxation times: The example of ^1H magnetization exchange in articular cartilage and hydrated collagen. *New J Phys* 2011;13:1–15.
- [4] Johnson C. Borehole nuclear magnetic resonance NMR. 2019, <http://dx.doi.org/10.5066/F73J3BW0>.
- [5] Mitchell J, Chandrasekera T, Gladden L. Numerical estimation of relaxation and diffusion distributions in two dimensions. *Prog Nucl Magn Reson Spectrosc* 2012;62:34–50.
- [6] Sun B, Dunn K-J. A global inversion method for multi-dimensional NMR logging. *J Magn Reson* 2005;172:152–60.
- [7] Borgia G, Bortolotti V, Brown R, Castaldi P, Fantazzini P, Soverini U. A comparison among different inversion methods for multi-exponential NMR relaxation data. *Magn Reson Imaging* 1994;12(2):209–12.
- [8] Testamanti MN, Rezaee R. Considerations for the acquisition and inversion of NMR T2 data in shales. *J Pet Sci Eng* 2019;174:177–88.
- [9] Zou Y, Li J, Hu S, Su J, Liu M, Zhang J. Two-dimensional NMR inversion based on fast norm smoothing method. *Energy Geosci* 2022;3(1):23–34.
- [10] L V, Yi-Qiao S, M.D. H. Solving Fredholm integrals of the first kind with tensor product structure in 2 and 2.5 dimensions. *IEEE Trans Signal Process* 2002;50:1017–26.
- [11] Enjilela R, Guo J, MacMillan B, Marica F, Afrough A, Balcom B. T1-T2* relaxation correlation measurements. *J Magn Reson* 2021;326:106961.
- [12] Yang D, Kausik R. 23NA and 1h NMR relaxometry of shale at high magnetic field. *Energy Fuels* 2016;30(6):4509–19.
- [13] Bortolotti V, Brown R, Fantazzini P. Upenwin. A software for inversion of multiexponential decay data for windows system. 2012, <http://software.dicam.unibo.it/upenwin>, URL <https://site.unibo.it/software/dicam/en/upenwin>.
- [14] Borgia G, Brown R, Fantazzini P. Uniform-penalty inversion of multiexponential decay data. *J Magn Reson* 1998;132(1):65–77. <http://dx.doi.org/10.1006/jmre.1998.1387>.
- [15] Borgia G, Brown R, Fantazzini P. Uniform-penalty inversion of multiexponential decay data: II. Data spacing, T2 data, systematic data errors, and diagnostics. *J Magn Reson* 2000;147(2):273–85. <http://dx.doi.org/10.1006/jmre.2000.2197>.
- [16] Bortolotti V, Brown RJS, Fantazzini P, Landi G, Zama F. Uniform penalty inversion of two-dimensional NMR relaxation data. *Inverse Problems* 2016;33(1):19.
- [17] Bortolotti V, Brizi L, Fantazzini P, Landi G, Zama F. Filtering techniques for efficient inversion of two-dimensional nuclear magnetic resonance data. *J Phys Conf Ser* 2017;904(1):012005.
- [18] Bortolotti V, Brown RJS, Fantazzini P, Landi G, Zama F. I2DUPEN: Improved 2DUPEN algorithm for inversion of two-dimensional NMR data. *Microporous Mesoporous Mater* 2018;269:195–8.
- [19] Bortolotti V, Landi G, Zama F. 2DNMR data inversion using locally adapted multi-penalty regularization. *Comput Geosci* 2021;25:1215–28. <http://dx.doi.org/10.1007/s10596-021-10049-y>.
- [20] Teal PD. Fast algorithms for estimation of NMR relaxation. 2016, URL <https://github.com/paultnz/flint>.
- [21] Modolo N, De Santi C, Minetto A, Sayadi L, Prechtl G, Meneghesso G, et al. Trap-state mapping to model gan transistors dynamic performance. *Sci Rep* 2022;12(1):1–10.
- [22] Blümich B. *Essential NMR*. Springer-Verlag; 2005.
- [23] Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J Imaging Sci* 2009;2(1):183–202. <http://dx.doi.org/10.1137/080716542>.
- [24] Fantazzini P, Galassi F, Bortolotti V, Brown RJ, Vittur F. The search for negative amplitude components in quasi-continuous distributions of relaxation times: The example of 1h magnetization exchange in articular cartilage and hydrated collagen. *New J Phys* 2011;13(6):065007.
- [25] Hair J, Black W, Babin B, Anderson R. *Multivariate data analysis*. 7th Ed.. New Jersey: Pearson Education International; Upper Saddle River; 2010.