JCTC Journal of Chemical Theory and Computation

Article

pubs.acs.org/JCTC

Inverse Laplace Transform of Multidimensional Relaxation Data Without Non-Negativity Constraint

Josef Granwehr* and Peter J. Roberts

Department of Physics and Astronomy, University of Nottingham, Nottingham NG7 2RD, United Kingdom

ABSTRACT: An algorithm based on Tikhonov regularization in generalized form is described to perform an inverse Laplace transform of multidimensional data without a non-negativity (NN) constraint for spectrum conditioning. Uniform penalty (UP) regularization is used to reduce the requirement for NN, and a further penalty is introduced for zero-crossing (ZC) of the spectrum. This ZC term is weighted with the slope of the curve, which does not prevent negative modes in the spectrum but makes nonphysical undershooting in the vicinity of narrow peaks more expensive. The performance of this algorithm is demonstrated using synthetic data, and the optimization of the free parameters for calculating the regularization matrix is discussed.

INTRODUCTION

When studying the time dependence of physical or chemical dynamic processes, it is found experimentally that many of these processes follow an exponential relaxation pattern.[1] If the molecular environment influences the relaxation behavior, but this environment does not average out quickly enough on the time scale of the relaxation-inducing interactions, it is common to observe a distribution of relaxation time constants.[2] Furthermore, if multiple different species are present, the situation may arise that the system is governed by more than one time constant. Many examples exist, including concentrations of reaction intermediates in chemical processes,[3] physical processes such as protein folding or isomerization,[4] or relaxation time constants in spectroscopic measurements.[5] To retrieve relaxation time constants from experimental data, suitable numerical methods are needed to fit a model to the data. For monoexponential decay or multiexponential decay with a known number of relaxation components, least-squares fitting is a common technique.[6] However, it is often not possible to know the exact number of relaxation components or the pattern of their distribution *a priori*. In this case, inversion algorithms for the relaxation data that do not make assumptions regarding the shape or the number of relaxation components, but instead provide a density function of the underlying relaxation distribution, are a more suitable tool.[7] This inversion is ill-conditioned.[8] Due to measurement noise, the source data can be represented by a wide variety of relaxation time distributions. It is therefore necessary to include prior knowledge to obtain a physical result in this case. A common assumption is that relaxation data are relatively smooth, which can be achieved numerically using a regularization term in the matrix inversion algorithm. Tikhonov regularization is one common algorithm for this purpose.[9] Further constraints are often applied to prevent oscillating solutions, such as a non-negativity (NN) constraint. NN assumes that all of the relaxation components have the same sign, which is not always physically justified. If cross-relaxation or exchange[10,11] occurs in the studied system, relaxation components with a negative sign will be suppressed by the constraint, and the density function obtained after inversion shows artificial features that are not connected to any physical relaxation process.[12,13] It was suggested that the extent of the additional fitting error be used when using NN to assess the validity of a negative peak. A significantly worse fit indicates that using NN is not warranted.[11]

For one-dimensional (1D) data sets, the uniform penalty (UP) method has been developed to reduce the need for a NN constraint and to provide good inversion results for samples with both narrow and broad relaxation features.[14−16] The regularization matrix is constructed iteratively using the first and the second derivatives of the density function from the previous iteration.

To understand the dynamics of a system, ideally one is looking for each time constant in the relaxation matrix that links the different states of a system.[17] However, complications arise if it is not possible to follow the relaxation processes individually on a microscopic level but only the overall concentration of the involved reaction partners. In this case, only the eigenvalues of the relaxation matrix are experimentally accessible. In nuclear magnetic resonance (NMR), recording multidimensional relaxation or diffusion maps, in combination with a two-dimensional (2D) inverse Laplace transform, has proved very successful in obtaining relaxation maps from contributions that would overlap when studied with a 1D measurement,[18] or in situations with multiple independent liquids in different environments, such as water and oil in a porous rock.[19]

The commonly used inversion algorithm[20] is based on the compression of the source data to reduce the size of the involved matrices, and a NN constrained inverse Laplace transform where the regularization parameter is optimized iteratively. In this article, an implementation of the UP regularization for n dimensions is presented. To reduce the need for a NN constraint, an additional penalty for zero-crossing of the density function is used. The different adjustable parameters are described, and suggestions for their adjustment are given. Finally,

Special Issue: Wilfred F. van Gunsteren Festschrift

Received: February 15, 2012
Published: April 16, 2012

dx.doi.org/10.1021/ct3001393 | *J. Chem. Theory Comput.* 2012, 8, 3473−3482

the performance of the algorithm is demonstrated on synthetic data.

THEORY

Problem Description. Multidimensional experimental data in R dimensions are commonly recorded as an array $S_R \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_R}$ of $N = \prod_{r=1}^{R} N_r$ measurements at different sampling points t_r, $r = 1, ..., R$, which are often evolution times but can be any variable parameter in an experiment, such as excitation frequency, magnetic field, or field gradient strength in the case of magnetic resonance. The signal is related to an underlying joint distribution function $\mathcal{G}(\tau_1, ..., \tau_R)$ of variables $\tau_1, ..., \tau_R$ by a Fredholm integral of the first kind:

$$s_R(t_1, ..., t_R) = \int_0^\infty ... \int_0^\infty k_1(t_1, \tau_1) \cdots k_R(t_R, \tau_R) \times \mathcal{G}(\tau_1, ..., \tau_R)\, d\tau_1 \cdots d\tau_R + \varepsilon(t_1, ..., t_R) \quad (1)$$

The kernels k_r are known continuous functions. $\varepsilon(t_1,...,t_R)$ represents an additive noise contribution, which is assumed to be independent and identically distributed (iid) Gaussian white noise with zero mean and variance σ^2. In the following, we scale the signal, without a loss of generality, by a factor σ^{-1} to obtain unit variance. Non-iid noise can be accounted for by scaling the signal and the kernels with the inverse of the noise variance.[15] Solving Fredholm integrals with smooth kernels, i.e., estimating $\mathcal{G}(\tau_1, ..., \tau_R)$ from $s_R(t_1, ..., t_R)$, is an ill-conditioned problem.

To represent eq 1 in matrix form, let $\mathbf{K}_r \in \mathbb{R}^{N_r \times M_r}$ denote the discretized kernel k_r, with number of elements M_r along dimension τ_r, and $G \in \mathbb{R}^{M_1 \times M_2 \times \cdots \times M_R}$ the discretized distribution function $\mathcal{G}(\tau_1,...,\tau_R)$, called the "spectrum" in the following. The total number of elements of the spectrum is $M = \prod_{r=1}^{R} M_r$. The signal vector

$$\mathbf{s} = \mathbf{K}\mathbf{g} + \mathbf{e} \quad (2)$$

where $\mathbf{s} \in \mathbb{R}^{N \times 1}$ is obtained by vectorizing S_R, $\mathbf{g} \in \mathbb{R}^{M \times 1}$ is the vectorization of G, $\mathbf{e} \in \mathbb{R}^{N \times 1}$ is the vectorization of the error matrix $E \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_R}$, and

$$\mathbf{K} = \mathbf{K}_R \otimes \mathbf{K}_{R-1} \otimes ... \otimes \mathbf{K}_1 \quad (3)$$

with $\mathbf{K} \in \mathbb{R}^{N \times M}$. $\otimes$ symbolizes the tensor product. Equation 2 reduces the problem of finding an R-dimensional distribution function G to a 1D problem.[18] At the end of the calculation, G can be obtained by undoing the vectorization process. However, the direct implementation of this algorithm is computationally expensive, as for typical data sets $\mathbf{K}$ becomes very large, and only moderately large 2D problems can realistically be solved on a desktop computer.

Data Compression. To reduce the matrix size and hence the memory requirement, Venkataramanan et al. used the tensor-product structure of the kernel to compress the data using a singular value decomposition (SVD).[20] Only singular values that contribute to the signal above the noise level are retained. The SVD of kernel $\mathbf{K}_r$ is

$$\mathbf{K}_r = \mathbf{U}_r \Sigma_r \mathbf{V}_r^{\mathrm{T}} \quad (4)$$

where $\mathbf{U}_r \in \mathbb{R}^{N_r \times N_r}$ and $\mathbf{V}_r \in \mathbb{R}^{M_r \times M_r}$ are unitary matrices and $\mathbf{\Sigma}_r \in \mathbb{R}^{N_r \times M_r}$ is a rectangular diagonal matrix containing the singular values in decreasing order along the diagonal. For smooth kernels $\mathbf{K}_r$, the singular values decay quickly to zero; hence, it is sufficient to project the data onto a much smaller subspace spanned by the n_r largest singular vectors.[21] The number n_r is selected as the smallest index that meets

$$\frac{(\Sigma_r)_1}{(\Sigma_r)_{n_r}} \leq \frac{s_{\max}}{\sigma} \quad (5)$$

i.e., by choosing the smallest submatrix with a condition number that is larger than the signal-to-noise ratio (SNR), defined as the maximum signal $s_{\max}$ divided by the standard deviation of the data.

The R-dimensional source data can be compressed along dimension r by first cyclically permuting the dimensions of the array S such that r is the first dimension, then matricizing the array such that the second to the Rth dimension are combined into the row of the generated matrix $\mathbf{S}_r \in \mathbb{R}^{N_r \times (\prod_{i \neq r} N_i)}$,[22] and eventually calculating

$$\tilde{\mathbf{S}}_r = \tilde{\mathbf{U}}_r^{\mathrm{T}} \mathbf{S}_r \quad (6)$$

The matrix $\tilde{\mathbf{U}}_r \in \mathbb{R}^{N_r \times n_r}$ is obtained as a submatrix of $\mathbf{U}_r$ by selecting only the first n_r columns of $\mathbf{U}_r$, i.e., $\tilde{\mathbf{U}}_r = \mathbf{U}_r[1, ..., N_r; 1, ..., n_r]$. The matricization along the row of the compressed matrix $\tilde{\mathbf{S}}_r \in \mathbb{R}^{n_r \times (\prod_{i \neq r} N_i)}$ and the cyclic permutation of the dimension can now be undone to obtain an array with source data compressed along dimension r, and the same procedure can be repeated for all dimensions that are to be compressed. Eventually, the compressed array $\tilde{S} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_R}$ is obtained. The decompressed data can be obtained by performing the same procedure, but using the inverse of the unitary projection matrix

$$\tilde{\tilde{\mathbf{S}}}_r = \tilde{\mathbf{U}}_r \tilde{\mathbf{S}}_r \quad (7)$$

To verify that a large enough number of singular values was selected, the Frobenius norm of the residuals between compressed and original data, $\|\tilde{\tilde{\mathbf{S}}}_r - \mathbf{S}_r\|_2$, should correspond to the norm of the random noise with known variance. Furthermore, the norm of these residuals should change only marginally for compression along different dimensions. The compressed kernels $\tilde{\mathbf{K}}_r \in \mathbb{R}^{n_r \times M_r}$ are obtained via

$$\tilde{\mathbf{K}}_r = \tilde{\Sigma}_r \tilde{\mathbf{V}}_r^{\mathrm{T}} \quad (8)$$

with the submatrices $\tilde{\Sigma}_r = \mathbf{\Sigma}_r[1, ..., n_r; 1, ..., n_r] \in \mathbb{R}^{n_r \times n_r}$ and $\tilde{\mathbf{V}}_r = \mathbf{V}_r[1, ..., M_r; 1, ..., n_r] \in \mathbb{R}^{M_r \times n_r}$.

The noise can be compressed and decompressed in the same way as the signal. Although decompression does not retrieve the original noisy data, compression or decompression does not change the noise statistics. The compressed noise, $\tilde{E} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_R}$, retains the mean and variance of E.

Equation 2 can now be rewritten using compressed matrices as

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

where $\tilde{\mathbf{s}} \in \mathbb{R}^{N_c \times 1}$, $\tilde{\mathbf{K}} = \tilde{\mathbf{K}}_R \otimes \tilde{\mathbf{K}}_{R-1} \otimes ... \otimes \tilde{\mathbf{K}}_1 \in \mathbb{R}^{N_c \times M}$, and $\tilde{\mathbf{e}} \in \mathbb{R}^{N_c \times 1}$ are obtained in the same way from $\tilde{\mathbf{S}}$, $\tilde{\mathbf{K}}_r$, and $\tilde{\mathbf{E}}$, respectively, as their uncompressed counterparts, and $N_c = \prod_{r=1}^{R} n_r$ is the total number of compressed data points.

Regularization. Equation 9 is ill-conditioned, so a direct inversion does not give a stable solution. Regularization is one possibility to implement a stable inversion of noisy data. Thereby a penalty term is added, and the functional to be minimized is

$$\hat{\mathbf{g}} = \underset{\mathbf{g}}{\operatorname{argmin}} \{ \| \tilde{\mathbf{K}}\mathbf{g} - \tilde{\mathbf{s}} \|_2^2 + \lambda^2 \| \mathbf{\Lambda}\mathbf{g} \|_2^2 \} \quad (10)$$

dx.doi.org/10.1021/ct3001393 | J. Chem. Theory Comput. 2012, 8, 3473−3482

where $\mathbf{\Lambda} \in \mathbb{R}^{M\times M}$ is the regularization matrix, and λ is a global scaling factor for the regularization matrix. Common choices for $\mathbf{\Lambda}$ are the identity matrix $\mathbf{I}_M \in \mathbb{R}^{M\times M}$, the first derivative operator $\mathbf{L}_1 \in \mathbb{R}^{M\times M}$, and the second derivative operator $\mathbf{L}_2 \in \mathbb{R}^{M\times M}$.[7] In NMR, $\mathbf{\Lambda} = \mathbf{I}_M$ in combination with a NN constraint for $\mathbf{g}$ is widespread.[9,23] For certain kernels, it is possible to drop the NN constraint and still get physically meaningful results,[24] but in general a different regularization strategy needs to be applied when NN is not justified. Borgia et al. proposed the UP regularization to improve the ability of the algorithm to correctly obtain relaxation spectra with both narrow and wide features.[14,15] Incidentally, UP also reduces the necessity to employ a NN constraint. In a multidimensional framework, the functional to be minimized for UP regularization, including the possibility to define constraints or additional penalties, can be written as

$$\hat{\mathbf{g}} = \underset{\mathbf{g}}{\operatorname{argmin}}\{\mathbf{g}^{\mathrm{T}}\tilde{\mathbf{K}}^{\mathrm{T}}\tilde{\mathbf{K}}\mathbf{g} - 2\tilde{\mathbf{s}}^{\mathrm{T}}\tilde{\mathbf{K}}\mathbf{g} + \tilde{\mathbf{s}}^{\mathrm{T}}\tilde{\mathbf{s}} + \lambda^2\mathbf{g}^{\mathrm{T}}\mathbf{\Lambda}^{\mathrm{T}}\mathbf{\Lambda}\mathbf{g}\} \tag{11}$$

If the spectrum only contains features that are either roughly circular or elongated along one of the axes, the regularization term $\mathbf{\Lambda}^{\mathrm{T}}\mathbf{\Lambda}$ can be composed of three different parts, each with a separate contribution for the different dimensions r:

$$\mathbf{\Lambda}^{\mathrm{T}}\mathbf{\Lambda} = \left(\sum_{r=1}^{R}\mathbf{A}_r\right)^{\mathrm{T}}\left(\sum_{r=1}^{R}\mathbf{A}_r\right) + \left(\sum_{r=1}^{R}\mathbf{D}_r\mathbf{L}_{1^*,r}\right)^{\mathrm{T}}\left(\sum_{r=1}^{R}\mathbf{D}_r\mathbf{L}_{1^*,r}\right) + \left(\sum_{r=1}^{R}\mathbf{C}_r\mathbf{L}_{2,r}\right)^{\mathrm{T}}\left(\sum_{r=1}^{R}\mathbf{C}_r\mathbf{L}_{2,r}\right) \tag{12}$$

$\mathbf{A}_r \in \mathbb{R}^{M\times M}$ is the coefficient matrix for amplitude regularization, $\mathbf{D}_r \in \mathbb{R}^{M\times M}$ is the coefficient matrix for slope regularization, and $\mathbf{C}_r \in \mathbb{R}^{M\times M}$ is the coefficient matrix for curvature regularization, which is the main UP term. All three matrices are diagonal. Constraints are not applied directly to the spectrum, but they are implemented as contributions to the regularization via the amplitude, slope and curvature terms.[15]

The coefficients of $\mathbf{C}_r$ must be determined separately for each direction r. $\mathbf{L}_{2,r}$ determines the second derivative along direction r only. It is obtained via

$$\mathbf{L}_{2,r} = -2\mathbf{I}_M + \mathbf{L}_{+,r} + \mathbf{L}_{-,r} \tag{13}$$

where

$$\mathbf{L}_{\pm,r} = \mathbf{I}_{M_R} \otimes \mathbf{I}_{M_R-1} \otimes \ldots \otimes \mathbf{L}_{\pm,(M_r)} \otimes \mathbf{I}_{M_r-1} \otimes \ldots \otimes \mathbf{I}_{M_1} \tag{14}$$

$\mathbf{L}_{+,(M_r)} \in \mathbb{R}^{M_r\times M_r}$ and $\mathbf{L}_{-,(M_r)} \in \mathbb{R}^{M_r\times M_r}$ are the first upper and lower subdiagonals, respectively, with elements $(L_{+,(M_r)})_{l,l+1} = (L_{-,(M_r)})_{l+1,l} = 1$ and $l = 1, \ldots, M_r - 1$. The first derivative operators $\mathbf{L}_{1,r}$ are obtained analogously. For the numerical implementation, three versions of $\mathbf{L}_{1,r}$ will be distinguished: $\mathbf{L}_{1^+,r} = -\mathbf{I}_M + \mathbf{L}_{+,r}$, then $\mathbf{L}_{1^-,r} = -\mathbf{L}_{-,r} + \mathbf{I}_M$, and eventually $\mathbf{L}_{1^*,r} = (-\mathbf{L}_{-,r} + \mathbf{L}_{+,r})/2$.

The elements of the coefficient matrix for the UP regularization are defined as[14]

$$(\mathbf{C}_r)_{k,l} = \frac{\delta_{kl}}{\sqrt{\alpha_{0,r}\Delta_r^5 + \alpha_{p,r}\Delta_r \max\left\{\left((\mathbf{L}_{1^*,r}\mathbf{g})^{(2)}\right)_j\right\}_{j=k-\eta_1,\ldots,k+\eta_1} + \alpha_{c,r}\Delta_r^{-1}\max\left\{\left((\mathbf{L}_{2,r}\mathbf{g})^{(2)}\right)_j\right\}_{j=k-\eta_2,\ldots,k+\eta_2}}} \tag{15}$$

where $(\cdot)^{(2)}$ represents the entry-wise square of a vector, δ_{kl} is the Kronecker delta function, Δ_r is the spacing between data points along the r direction of the spectrum, and $\eta_{1,2}$ specifies the number of points over which the maximum is determined for each term in the regularization matrix along direction r. Typically, we chose $\eta_{1,2} = 1$.[14] For logarithmic spacing of the data points in the spectrum, which is the common choice for this type of inversion, $\Delta_r = \ln(\tau_{r,2}) - \ln(\tau_{r,1})$ is in units of neper. $\alpha_{0,r}$, $\alpha_{p,r}$, and $\alpha_{c,r}$ are adjustable compliance parameters for the floor, the slope, and the curvature, respectively. They are, in general, different for each dimension r of the spectrum but unchanged during the iterative process of determining a minimum solution for the object function. When calculating the local maximum over $2\eta + 1$ neighboring points of a vector, $\max\{\cdot\}_{j=k-\eta,k-\eta+1,\ldots,k+\eta}$, the values in the argument corresponding to $j < 1$ or $j > M_r$ were set to zero. If the spectrum contains a static offset along dimension r that represents the value for $\tau_r \rightarrow \infty$, this data point is only affected by the $\alpha_{0,r}$ term in $\mathbf{C}_r$.

The scaling of the different terms in the denominator on the right-hand side of eq 15 was chosen to keep $\alpha_{0,r}$, $\alpha_{p,r}$,and $\alpha_{c,r}$ independent of M_r. If we require $\lambda^2\|\mathbf{\Lambda}\mathbf{g}\|_2^2$ to be invariant upon changing M_r, we can use the fact that $\Delta_r \propto M_r^{-1}$ for fixed axis limits $\tau_{r,\min}$ and $\tau_{r,\max}$ to balance $\mathbf{C}_r$. Since we do not scale kernels as a function of M_r, $\mathbf{g}$ scales approximately proportionally to M_r^{-1}. Applying $\mathbf{L}_{1,r}$ or $\mathbf{L}_{2,r}$ causes a scaling of the result proportional to M_r^{-1} or M_r^{-2}, respectively. Finally, the number of elements in the scalar product $(\mathbf{\Lambda}\mathbf{g})^{\mathrm{T}}(\mathbf{\Lambda}\mathbf{g})$ scales proportionally to M_r; hence, $\mathbf{C}_r \propto M_r^{5/2}$ or, alternatively, $\mathbf{C}_r \propto \Delta_r^{-5/2}$.

Although UP considerably reduces the dependence on a NN constraint, sharp lines can still cause substantial undershooting. To make such undershoots more expensive, an additional zero-crossing (ZC) penalty is imposed, with the diagonal coefficient matrix

$$(\mathbf{D}_r)_{k,l} = \frac{\max\left\{\left(\left(\frac{1}{2}\mathbf{L}_{1^-,r}\boldsymbol{\zeta}_g\right)^{(2)} + \left(\frac{1}{2}\mathbf{L}_{1^+,r}\boldsymbol{\zeta}_g\right)^{(2)}\right)_j\right\}_{j=k-\eta_D,\ldots,k+\eta_D}\delta_{kl}}{\sqrt{\alpha_{d,r}\Delta_r^3 + \alpha_{D,r}\Delta_r^{-1}((\mathbf{L}_{1^*,r}\mathbf{g})^{(2)})_k}} \tag{16}$$

where $\boldsymbol{\zeta}_g \in \mathbb{R}^{M\times 1}$ represents the entry-wise sign of $\mathbf{g}$, i.e., $(\boldsymbol{\zeta}_g)_k = \operatorname{sign}(\mathbf{g}_k)$. In the denominator, the parameter $\alpha_{d,r}$ determines the weighting and $\alpha_{D,r}$ sets an upper limit for the ZC penalty. Such an upper limit is important to prevent this penalty from dominating the regularization matrix. Alternatively, an upper limit can be set by capping $(\mathbf{D}_r)_{k,l}$ at a multiple δ of the corresponding value $(\mathbf{C}_r)_{k,l}$ of the UP coefficient matrix. δ is typically on the order of unity. The value η_D was usually set to zero for coarsely sampled spectra, while

dx.doi.org/10.1021/ct3001393 | J. Chem. Theory Comput. 2012, 8, 3473−3482

an η_D of 1 or 2 helped to stabilize the spectrum and hence improved convergence of the algorithm for densely sampled spectra.

For a robust inversion, it is necessary that the minimum $\tau_{r,\min}$ and the maximum $\tau_{r,\max}$ of the r axis are chosen well beyond the respective minimum and maximum signal of the expected spectrum. Artifacts at the boundaries can be prevented by adding an additional penalty, employing an amplitude regularization for the first and last $h_{0,r}$ points along dimension r,[15]

$$(\mathbf{A}_r)_{k,l} = \frac{(\mathbf{H}_r)_{k,l}}{\sqrt{\alpha_{a,r}\Delta_r + \alpha_{A,r}\Delta_r^{-1}(\mathbf{g}^{(2)})_k}} \tag{17}$$

where $\alpha_{a,r}$ determines the weighting and $\alpha_{A,r}$ sets an upper limit for the regularization coefficient of a particular data point. The mask $\mathbf{H}_r \in \mathbb{R}^{M\times M}$ is a diagonal matrix defined by

$$\mathbf{H}_r = \mathbf{I}_{M_R} \otimes \mathbf{I}_{M_{R-1}} \otimes \ldots \otimes \mathbf{H}_{0,r} \otimes \mathbf{I}_{M_{r-1}} \otimes \ldots \otimes \mathbf{I}_{M_1} \tag{18}$$

with

$$(\mathbf{H}_{0,r})_{k,l} = (H(h_{0,r} - k) + H(h_{0,r} - M_r + k))\delta_{kl} \tag{19}$$

where $H(\cdot)$ denotes the Heaviside function.

If a spectrum contains elongated features along the diagonal or antidiagonal direction, the regularization matrix as described by eq 12 is not fully adequate. The reason for this is the inability of first and second derivative operators along the axial directions to predict the slope and curvature, respectively, of these spectral features. While the lack of a ZC penalty along the diagonal does not significantly alter the result, the UP regularization tends to break open elongated lines into multiple peaks. At the same time the norm of the residuals increases. Significant over-regularization would be necessary to prevent this problem. Alternatively, the curvature term $\mathbf{\Lambda}_{\text{up}}^{\text{T}}\mathbf{\Lambda}_{\text{up}}$ of eq 12, i.e., the regularization matrix without contributions from the ZC penalty or the amplitude regularization at the boundaries, can be extended to also include the second derivative along each of the diagonal and antidiagonal directions. For the 2D case, or if diagonal and antidiagonal features occur between no more than two dimensions at a time, this can be written as

$$\mathbf{\Lambda}_{\text{up}} = \sum_{r=1}^{R}\sum_{s=1}^{R} \mathbf{C}_{rs}\mathbf{L}_{2,rs} \tag{20}$$

Thereby, the direction shall be parallel to one of the axes for $r = s$, i.e., $\mathbf{C}_{rr} = \mathbf{C}_r$ and $\mathbf{L}_{2,rr} = \mathbf{L}_{2,r}$, diagonal for $r < s$, and antidiagonal for $r > s$. The corresponding timing value is $\Delta_{rs} = (\Delta_r^2 + \Delta_s^2)^{1/2}$, and the different α_x values are calculated as the root-mean-square (RMS) of the corresponding axial terms:

$$\alpha_{x,rs} = \sqrt{\frac{\alpha_{x,r}^2 + \alpha_{x,s}^2}{2}} \tag{21}$$

where x represents any of the subscripts used for α. The operators $\mathbf{L}_{2,rs}$ for $r \neq s$ do not have tensor product structure but can be constructed as the sum of a set of operators with tensor product structure, each of which represents one of the diagonals in the second derivative operator. For the 2D case, we get

$$\mathbf{L}_{2,12} = -2\mathbf{I}_M + \mathbf{L}_{+,M_2} \otimes \mathbf{L}_{+,M_1} + \mathbf{L}_{-,M_2} \otimes \mathbf{L}_{-,M_1} \tag{22}$$

for the diagonal direction, and

$$\mathbf{L}_{2,21} = -2\mathbf{I}_M + \mathbf{L}_{+,M_2} \otimes \mathbf{L}_{-,M_1} + \mathbf{L}_{-,M_2} \otimes \mathbf{L}_{+,M_1} \tag{23}$$

for the antidiagonal direction. Correspondingly, $\mathbf{L}_{1^*,12} = (-\mathbf{L}_{-,M_2}\otimes \mathbf{L}_{-,M_1} + \mathbf{L}_{+,M_2}\otimes\mathbf{L}_{+,M_1})/2$ and $\mathbf{L}_{1^*,21} = (-\mathbf{L}_{+,M_2}\otimes\mathbf{L}_{-,M_1} + \mathbf{L}_{-,M_2}\otimes\mathbf{L}_{+,M_1})/2$. For higher dimensions, in-plane diagonals can be constructed analogously, but dimensions that are not part of the diagonal must be represented by unity matrices as in eq 14.

Parameter Selection. To solve eq 10, several user-defined parameters must be adjusted. The most important ones are the parameters $\alpha_{0,rs}$, $\alpha_{p,rs}$ and $\alpha_{c,rs}$ for the UP regularization. $\alpha_{0,rs}$ is a compliance floor that should be small enough for the algorithm not to cause undersmoothing but large enough to seed the development of curvature in the iterative execution of the algorithm.[14] This term is only dominant in regions with very low or no spectral density for all but very low SNR data. Its exact value is not critical, but it is usually selected several orders of magnitude smaller than the other terms contributing to the denominator of $\mathbf{C}_{rs}$. In our simulations, we used $\alpha_{0,rs} = 10^{-4}$.

The main term, which ensures the penalty to be uniform, is the curvature term with parameter $\alpha_{c,rs}$. For a good solution, the norm of the regularization term should be comparable to the norm of the residuals.[14,25] We determine α_c by setting $\alpha_0 = 0$ and $\alpha_p = 0$, and equating the norms of regularization and residuals

$$\|\tilde{\mathbf{K}}\mathbf{g} - \tilde{\mathbf{s}}\|_2 = \lambda\|\mathbf{\Lambda}\mathbf{g}\|_2 \tag{24}$$

Knowing that the noise has unity variance, this equation can be rewritten as

$$\lambda^2\|\mathbf{\Lambda}\mathbf{g}\|_2^2 = \prod_{r=1}^{R} n_r \tag{25}$$

To obtain an initial guess, let us make the idealized assumptions that, on average, the curvatures along the different dimensions are not correlated, that the regularization matrix is dominated by the UP term, and that eq 12 provides an adequate initial guess. Then we get

$$\lambda^2\mathbf{g}^{\text{T}}\left[\left(\sum_{r=1}^{R}\mathbf{C}_r\mathbf{L}_{2,r}\right)^{\text{T}}\left(\sum_{r=1}^{R}\mathbf{C}_r\mathbf{L}_{2,r}\right)\right]\mathbf{g} \sim \sum_{r=1}^{R}\frac{M\Delta_r}{\alpha_{c,r}} \tag{26}$$

By default, we set $\lambda = 1$. If we further require that each dimension contributes the same amount to the UP regularization term and equate eqs 25 and 26, we get

$$\alpha_{c,r} = R\Delta_r\prod_{\rho=1}^{R}\frac{M_\rho}{n_\rho} \tag{27}$$

The slope term $\alpha_{p,r}$ in the UP regularization prevents unphysical straight segments in the spectrum, and it avoids the breaking up of broad lines into two or more very narrow lines. The exact ratio $\alpha_{p,r}/\alpha_{c,r}$ is not critical. It was suggested to choose a fixed ratio of $\alpha_{p,r}/\alpha_{c,r}$.[15] Using $\alpha_{p,r}/\alpha_{c,r} = 5$ gave good results in the simulations used to test this method.

Notice that with the selection of $\alpha_{c,r}$ and $\alpha_{p,r}$ using the above procedure, the equality described by eq 24 is not fulfilled any more. To obtain a set of parameters that is closer to this target value, a first fit with the above parameters is performed. Using the resulting spectrum, all α_c and α_p values are uniformly scaled by $\|\mathbf{\Lambda}_{\text{up}}\mathbf{g}\|_2/\|\tilde{\mathbf{K}}\mathbf{g} - \tilde{\mathbf{s}}\|_2$. Fine-tuning of the regularization matrix is subsequently done by adjusting λ.

The default value for the global scaling factor was set to $\lambda = 1$. This choice was based on eq 24, and the fact that $\alpha_{c,r}$ was selected to meet this condition. However, the other contributions to $\mathbf{\Lambda}$ can cause a deviation of the residual norm from this equality,

which can be adjusted by changing λ. Optimization of the parameter λ can be done using one of the established techniques for Tikhonov regularization.[7] We usually employed either the S-curve method[23] or a modification of the L-curve method. For the S-curve method, $\log(\|\mathbf{Kg} - \mathbf{s}\|_2)$ is plotted as a function of $\log(\lambda)$. An optimum choice of λ is found at the heel of the curve when $\|\mathbf{Kg} - \mathbf{s}\|_2$ approaches a plateau close to its minimum value. This plateau is typically slightly lower than the norm of the random noise. One can use the point where the residual norm crosses the level set by the noise norm as the optimum choice for λ. For the L-curve method, spectra are calculated for different values of λ. The results are plotted, on a log–log scale, as solution norm $\|\mathbf{g}\|_2$ vs residual norm $\|\mathbf{Kg} - \mathbf{s}\|_2$ of the different fits. The optimal choice of λ causes a result near the corner of the obtained L-shaped curve. For synthetic as well as for experimental data, we usually obtained an optimum value λ_{opt} that was somewhat smaller than unity. The typical value was $\lambda_{opt} \approx 0.3–0.5$.

Certain spectral features could be reproduced more accurately by carefully adjusting the different parameters in the UP algorithm relative to each other. However, this does require the inclusion of some prior knowledge regarding the expected lineshapes, and it does not lead to significantly better fits of the experimental data in most cases. This seemingly counterintuitive behavior was previously explained for the 1D case,[15] and it equally applies for multiple dimensions. The slope and the curvature compliance contributions cause a similar adjustment of the regularization matrix. Both are dominant in regions of the spectrum where a signal can be observed, while their effect is small in regions without signal. However, within each line, they primarily affect different sections. As long as both parameters are adjusted to contribute a similar order of magnitude to the regularization matrix, obvious artifacts in the lineshapes can be avoided, irrespective of the exact ratio $\alpha_{p,r}/\alpha_{c,r}$. In regions with no or a very small signal, the $\alpha_{0,r}$ parameter is dominant. This parameter only has a minor effect for data with a reasonably high SNR; hence, its exact value does not drastically change the obtained spectrum. Finally, the aim of the ZC penalty is to prevent oscillating solutions that are not supported by the data. If chosen too small, its effect is negligible, while a too large ZC penalty dominates the regularization matrix. However, the range of the parameters $\alpha_{d,r}$ and $\alpha_{D,r}$, where meaningful results are obtained, is relatively large; thus the exact choice of these parameters is not critical either. Nonetheless, one of the main issues of the presented regularization algorithm is the large number of adjustable parameters. Therefore, it is important that these are selected in a reproducible and algorithmic way to minimize the necessary user intervention.

Algorithm for Spectrum Calculation. The object function, eq 10, is solved iteratively until convergence is achieved. Coefficient matrices $\mathbf{A}_r$, $\mathbf{D}_r$, and $\mathbf{C}_{rs}$ for iteration i are calculated using $\hat{\mathbf{g}}_{i-1}$. For $i = 1$, vector $\hat{\mathbf{g}}_0$ is set to be the null vector. To invert the regularized equation, different methods are possible. Direct inversion

$$\hat{\mathbf{g}} = (\tilde{\mathbf{K}}^T\tilde{\mathbf{K}} + \lambda^2\mathbf{\Lambda}^T\mathbf{\Lambda})^{-1}\tilde{\mathbf{K}}^T\tilde{\mathbf{s}} \tag{28}$$

is only feasible for spectra with a relatively small number of points. However, for 1D and small 2D spectra, it is often the fastest method.

For larger matrices, we can rewrite the generalized Tikhonov regularization in standard form with the identity matrix as the regularization matrix.[7] Then

$$\hat{\bar{\mathbf{g}}} = \underset{\bar{\mathbf{g}}}{\operatorname{argmin}}\{\|\bar{\mathbf{K}}\bar{\mathbf{g}} - \bar{\mathbf{s}}\|_2^2 + \lambda^2\|\bar{\mathbf{g}}\|_2^2\} \tag{29}$$

with $\bar{\mathbf{K}} = \tilde{\mathbf{K}}\bar{\mathbf{\Lambda}}^{-1}$ where $\bar{\mathbf{K}} \in \mathbb{R}^{N_c \times M}$, $\bar{\mathbf{g}} = \bar{\mathbf{\Lambda}}\mathbf{g}$ where $\bar{\mathbf{g}} \in \mathbb{R}^{M\times 1}$, and $\bar{\mathbf{s}} = \tilde{\mathbf{s}}$ for iid noise with unit variance, where $\bar{\mathbf{s}} \in \mathbb{R}^{N_c \times 1}$. $\bar{\mathbf{\Lambda}} \in \mathbb{R}^{M\times M}$ is obtained as the result of a Cholesky decomposition of $\mathbf{\Lambda}^T\mathbf{\Lambda}$. This decomposition can be performed efficiently due to the sparsity of matrix $\mathbf{\Lambda}^T\mathbf{\Lambda}$, for which the number of nonzero elements scales proportionally with M. Back-transformation of the result is done via

$$\hat{\mathbf{g}} = \bar{\mathbf{\Lambda}}^{-1}\hat{\bar{\mathbf{g}}} \tag{30}$$

For compressed source data, M is usually significantly larger than N_c. Hence, direct inversion of eq 29, analogous to eq 28 for the generalized form of the regularization, becomes expensive because an $M \times M$ matrix must be inverted. Using the Woodbury matrix identity,[26] the direct inversion can be rewritten as a problem that requires the inversion of an $N_c \times N_c$ matrix:

$$\hat{\bar{\mathbf{g}}} = \lambda^{-2}\bar{\mathbf{K}}^T\left[\mathbf{I}_{N_c} - \lambda^{-2}\left(\mathbf{I}_{N_c} + \lambda^{-2}\bar{\mathbf{K}}^T\bar{\mathbf{K}}\right)^{-1}\bar{\mathbf{K}}^T\bar{\mathbf{K}}\right]\bar{\mathbf{s}} \tag{31}$$

There are also various other methods available in the literature to solve eq 29.[7,20] For even larger problems, iterative solutions can be used to analyze the data.[7]

If prior knowledge suggests that a spectrum does not contain elongated features in diagonal or antidiagonal directions,[27] but that instead all of the modes are either along one of the axes or of roughly circular shape, eq 12 can be applied without the diagonal and antidiagonal terms in the regularization matrix. This increases the sparsity of the regularization matrix and therefore accelerates the inversion step. Circular or elliptically shaped features tend to adopt a slightly rectangular shape at their base, but the performance of the algorithm does not get considerably degraded.

A different approach would be to use the curvature regularization with a single coefficient λ, i.e., setting $\mathbf{C}_r = \mathbf{I}_M$. This would still lessen the necessity of a NN constraint compared to a standard-form Tikhonov regularization, while at the same time reduce the number of free parameters of the algorithm and lower the computation time. In this case, acceptable results could be obtained by setting $\eta_D > 1$, but more care is required to test the validity of the features in the spectrum.

Diagnostics. To ensure the validity of a fit with UP regularization and to identify possible problems with the data, various statistical tests can be performed.[15] Problems with fits of the experimental data can usually be spotted by calculating the mean and the variance of the residuals between the original data and the fitted data using all of the data points along one dimension and plotting the results along the other dimensions. If the resulting curve shows some features, the difference between fit and original data should be inspected to make sure there is no problem with the source data, such as outliers or nonuniform noise. If this is not the case, it is likely that the data were over-regularized, causing an oversmoothing of the features in the spectrum, and the inversion should be repeated with a reduced value of λ. If the mean along one dimension is not zero within the noise boundaries, it is also possible that the number of singular values used for data compression was not sufficiently large. If this was the case, the residuals between the source data and the decompressed data would show the same features as the mean. If the variance is uniform along all dimensions but shows a mean value that significantly deviates from unity, it is likely that the noise variance of the source data was under- or overestimated. For reproducible calculations, the data should be rescaled and the calculation repeated. Eventually, if the variance shows features which are correlated with the signal intensity

and which cannot be improved by adjusting the regularization parameters, the data are likely to be affected by multiplicative noise.[28] This problem becomes more pronounced for data with high SNR. If this occurs, the multiplicative noise level should be estimated and a covariance matrix calculated for the data. The inverse of the Cholesky factors of this covariance matrix can then be used for weighting the data and kernels.[7] For estimating the multiplicative noise level, it may be possible to use the residuals of an inverse Laplace transform iteratively.[15]

To identify under-regularization and the corresponding appearance of spurious features in the spectrum, the fit can be repeated multiple times with identical parameters by adding synthetic iid noise with unit variance to the original fitted data. From the results, a mean and standard deviation for the refitted spectrum can be calculated. If the spectrum cannot be refitted reproducibly or if the standard deviation of the spectrum is comparable in magnitude to the amplitude of the different modes, under-regularization is most likely the reason. The original fit should then be repeated with a larger value of λ. A spectrum that is not self-consistent, i.e., the original lies outside the error bars of the refits but the refits themselves are robust, points to over-regularization or to problems with the source data.

Another feature that indicates under-regularization, which can usually be spotted immediately, is the appearance of unphysically intense and narrow lines in the spectrum. This could also indicate that $\alpha_{p,rs}/\alpha_{c,rs}$ was chosen too small. Increasing $\alpha_{p,rs}$ or, alternatively, η_1 and η_2 can prevent this problem.

SIMULATIONS

Description of Simulation System. To assess the performance of the algorithm, simulations of a 2D data set with kernels $k_1(t_1, \tau_1) = \exp(-t_1/\tau_1)$ and $k_2(t_2, \tau_2) = \exp(-t_2/\tau_2)$ were performed.[29] A spectrum consisting of four lines with Gaussian shape in log-space, as shown in Figure 1a, was chosen. Two of the lines had a circular base, and two were elongated, one of them in an axial direction and the other one in the antidiagonal direction. The anti-diagonal line had a negative sign, while the other three were positive. The four lines had similar amplitudes, but different integrals. The integral of the narrow peak was about an order of magnitude lower than the integral of the three other peaks. The data points were spaced logarithmically between 10^{-4} s and 10 s in both dimensions. The data matrix calculated from the exact spectrum consisted of 300 × 100 elements. Gaussian white noise of unit variance was added. The fitted spectrum consisted of 60 logarithmically spaced data points between 10^{-4} s and 1 s in both dimensions. Unless explicitly stated, simulations for different SNRs were performed with standard fitting parameters $\lambda = 1$, $\alpha_0 = 10^{-4}$, $\alpha_{c,r}$ calculated using eq 27, $\alpha_{p,r} = 5\alpha_{c,r}$, and $\alpha_d = 10^{-5}$. The elements of the ZC penalty were capped at $\delta = 1$. λ was then optimized manually using the S-curve method. An amplitude penalty was applied over two points at the boundary of the spectrum in both dimensions.

Notice that with these relaxation kernels and logarithmically spaced time points of the spectrum, the spectra amplitudes are represented as signal per neper of relaxation time. Hence, the area under a line with logarithmic axis scaling of the spectrum is proportional to the number of contributing spins.[30]

Performance as a Function of SNR. Results for simulations with an SNR of 1000, 100, and 10 are shown in Figure 1b–d. The sum projection onto the τ_1 axis indicates the reproducibility of the linewidths for the different peaks. With an SNR of 1000, both widths and intensities were faithfully reproduced for the lines with a large integral. The narrow line

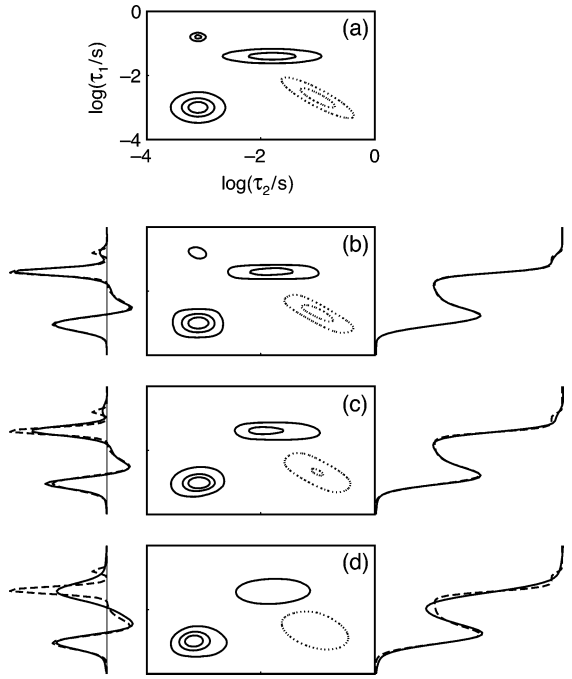

Figure 1. Inversion of 2D relaxation data for different SNR values. Contour levels were selected at 0.9, 0.5, and 0.1 times the signal maximum of the original spectrum. Solid contours represent peaks with a positive sign, dotted contours represent peaks with a negative sign. For each fitted spectrum, the sum projection along the τ_1 dimension is shown to the left and the cumulative sum is shown to the right of the contour plot. The solid line represents the respective spectrum and the dotted line the exact spectrum. (a) Contour plot of exact spectrum. (b) SNR of 1000. (c) SNR of 100. (d) SNR of 10.

with a small integral was broadened but clearly distinguished from the other peaks in the contour plot as well as in the sum projection. With an SNR of 100, the narrow line could still be distinguished from the other peaks, as seen in the cumulative sum and the sum projection, but the line was significantly broadened. The intense lines were correctly identified but slightly broadened. With an SNR of 10, the lines were broadened, but the main spectral features could be identified and the cumulative sum still provided a good result. The signs of all peaks were predicted correctly, and there were no spurious features in the spectra. None of the spectra showed noticeable oscillations that could cause a misinterpretation of the data.

Effect of ZC Penalty. Figure 2 shows the influence of the ZC penalty on the spectrum. With a moderate ZC penalty (Figure 2a), undershooting at the fringes of the lines can be minimized, while the spectrum can still show features with positive and negative signs. With no ZC penalty, undershooting is common mainly for high SNR and narrow lines, although this effect does not dominate the spectrum when using UP regularization (Figure 2b). Nonetheless, the undershooting becomes more significant and may cause features that could be misinterpreted as peaks with opposite sign. At lower SNR, the UP regularization prevents undershooting of the signal sufficiently well such that the ZC penalty would not be necessary. At the same time, the ZC penalty does not affect the result considerably; hence, the default parameters can be kept to reduce the number of variables that need to be adjusted.

Figure 2c shows the effect of a very large ZC penalty. It becomes the dominant regularization term and causes a significant under-regularization. The spectrum only consists of

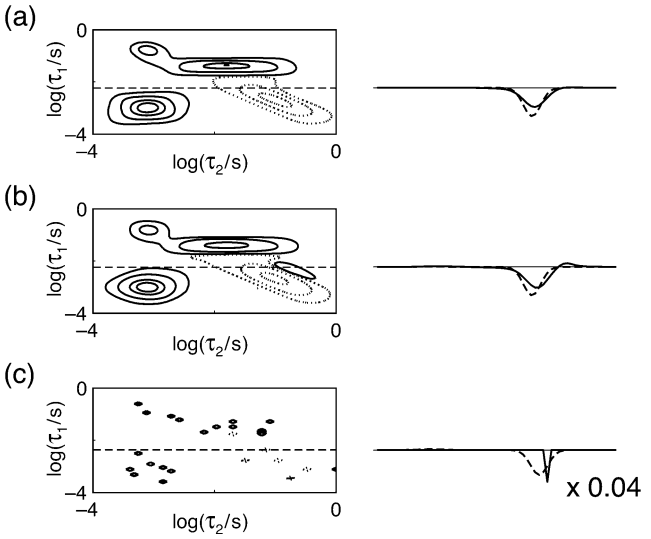

Figure 2. Influence of ZC penalty on a spectrum with an SNR of 1000. Contour levels were selected at 0.9, 0.5, 0.1, and 0.01 times the signal maximum of the original spectrum. The same data set as for Figure 1 was investigated. On the right, a trace across the spectrum corresponding to the horizontal line in the contour plot is shown. The dashed curve represents the same trace of the original spectrum. (a) Simulation with ZC penalty parameters $\alpha_{d,r} = 10^{-3}$ for all dimensions r and $\delta = 1$. (b) Simulation without ZC penalty. (c) Simulation with $\alpha_{d,r} = 10^{-10}$ and ZC penalty cap increased to $\delta = 10^4$. The trace is scaled by a factor of 0.04 compared to the original spectrum.

very narrow spikes, but it is not ill-conditioned yet. The spikes cover the parts of the spectrum where the lines are located in the original spectrum, and the correct signs are predicted, but the result displays a resolution that is not justified for the data.

Variation of Regularization Parameters. To demonstrate that the performance of the algorithm does not critically depend on the exact choice of the different α values relative to each other, Figure 3 shows the effect of changing one adjustable α parameter at a time by 1 order of magnitude above and below its default value in each dimension. In the upper panels, contour plots of the obtained spectra are overlaid. Overall, the variation of the spectral features is small. In particular, changing α_0 and α_d only affects minor details of the spectrum. The main difference could be observed for the narrow peak with a smaller integral than the other peaks. As would be expected for a signal with good SNR, changing the α_c and α_p parameters showed the biggest effect on the spectrum. It is important to notice that the two parameters must, to a certain extent, balance each other. They affect different parts of a line—α_c weighs strongest in its center and α_p at its inflection point. If α_p/α_c is too small, α_c becomes dominant. This tends to cause narrow lines to narrow further, until the signal eventually collapses into a single pixel (Figure 3a, blue line). This problem is usually accompanied by a very slow convergence of the spectrum.

The lower panels of Figure 3 show one trace from the matricized diagonal of $\lambda^2\mathbf{\Lambda}^T\mathbf{\Lambda}$. Varying the different α values does not significantly change the pattern of the regularization. Again, α_c and α_p have the largest effect. α_0 only influences the result in areas with no noticeable signal; hence, it has a small influence on the spectrum. α_d has the largest effect in the wings of the lines. The stronger the variation caused by this parameter, the more likely a line is to show over- or undershooting in the absence of the ZC penalty.

For all of these simulations, the norm of the residuals between fit and data was slightly below the norm of the added synthetic noise, i.e., $\|\mathbf{K}\hat{\mathbf{g}} - \mathbf{s}\|_2 < \|\mathbf{e}\|_2$, with $(\|\mathbf{e}\|_2 - \|\mathbf{K}\hat{\mathbf{g}} - \mathbf{s}\|_2)/\|\mathbf{e}\|_2 < 10^{-3}$. For comparison, this was considerably less than the variation of $\|\mathbf{e}\|_2$ for multiple repetitions of the simulation using different iid Gaussian white noise vectors with unit variance. Hence, a moderate variation of the individual α parameters from their default values did not significantly improve or worsen the fit of the data.

S-Curve Method. Figure 4 shows S-curves[23] for different SNR values. The optimum value of λ for a data set with known noise variance can be obtained as the point where the norm of the residuals equals the norm of the random noise, shown in the inset of the figure. Although the S-curve is not as smooth as for Tikhonov regularization in standard form with λ as the only variable parameter, the general character of the curve is retained, and the same method can be used to find an optimum value for λ.

Self-Consistency Test. To test the robustness of the algorithm, the data set with an SNR of 100 was fitted with nine different λ values, logarithmically spaced between $\lambda = 0.1$ and $\lambda = 10$. For each λ, the spectrum $\hat{\mathbf{g}}_\lambda$ and the corresponding signal $\mathbf{s}_\lambda$ were fitted. The fitted signal was then used with ten

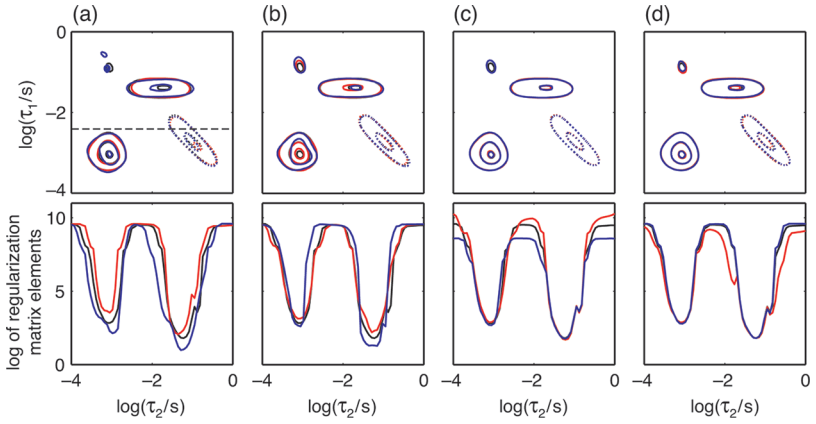

Figure 3. Spectra obtained by varying one free regularization parameter at a time in each dimension. Each parameter was set a factor 10 above (blue) and below (red) its default value (black). The data had an SNR of 120. The upper panels contain contour plots of the obtained spectra, with contour levels at 0.9, 0.5, and 0.1 times the maximum of the original spectrum. Dotted lines indicate negative contours. The lower panels show one trace of the matricized diagonal of $\lambda^2\mathbf{\Lambda}^T\mathbf{\Lambda}$, corresponding to the points represented by the dashed horizontal line in the top left panel. (a) Variation of α_c. (b) Variation of α_p. (c) Variation of α_0. (d) Variation of α_d.

dx.doi.org/10.1021/ct3001393 | J. Chem. Theory Comput. 2012, 8, 3473−3482

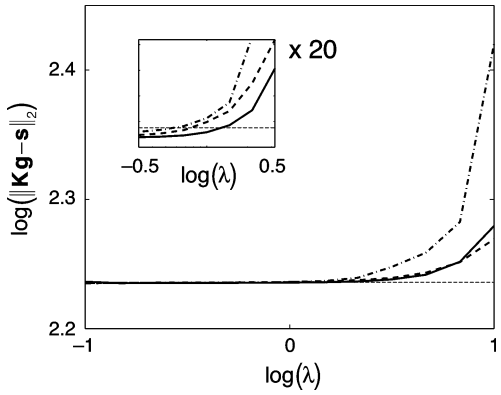

Figure 4. S-curve representation of the norm of the residuals as a function of the global scaling factor λ at SNRs of 1000 (solid), 100 (dashed), and 10 (dash-dotted). The inset highlights the region of optimal λ and is vertically scaled by a factor 20. The horizontal small-dashed line represents the norm of the random noise.

different noise vectors of unit variance to refit ten different spectra with identical parameters. These spectra were used to calculate a mean $\langle \mathbf{g}_{\text{refit},\lambda} \rangle$ and a standard deviation $\boldsymbol{\sigma}_{\text{refit},\lambda}$ for each point in the spectrum. Eventually, this data set was used to test how well the original fit could be reproduced from the refitted data. The results of this self-consistency test are shown in Figure 5. From Figure 5a, which shows the standard deviation for each point in the spectrum, it is evident that at positions where a peak is fitted by the algorithm, the standard deviation is much larger than at positions with no signal. The result of this calculation can be used to determine an SNR value for the refitted spectrum as the ratio of the norm of the spectrum relative to the norm of the standard deviation:

$$\psi_{\text{norm}}(\lambda) = \frac{\left\| \langle \mathbf{g}_{\text{refit},\lambda} \rangle \right\|_2}{\| \boldsymbol{\sigma}_{\text{refit},\lambda} \|_2} \tag{32}$$

This value is a measure of the robustness of a fit. Alternatively, the SNR can be calculated as the ratio of the maximum of the spectrum relative to the maximum of the standard deviation:

$$\psi_{\text{max}}(\lambda) = \frac{\max\{\langle \mathbf{g}_{\text{refit},\lambda} \rangle\}}{\max\{\boldsymbol{\sigma}_{\text{refit},\lambda}\}} \tag{33}$$

Since the standard deviation is roughly proportional to the signal, ψ_{norm} and ψ_{max} should have comparable values and follow similar trends. As shown in Figure 5b, this is achieved reasonably well for high values of λ. However, at $\lambda \approx 0.3$, the two curves start to deviate. This is caused by sharp, unphysical spikes appearing in a few of the refitted spectra. If the number of spikes is small, the influence on ψ_{norm} is limited, while ψ_{max} is affected much more strongly. Hence, the point of deviation of the two curves indicates the value of λ where the fit becomes unreliable. λ should be chosen higher than this value. For even smaller values of λ, when the inversion is significantly under-regularized, the spectra for all noise vectors become dominated by narrow spikes, and the two ψ values get closer again.

Another method to analyze the data is to compare the quality of the original fit with the quality of the refit (Figure 5c). Ideally, the two residuals should be very similar, with the refit slightly higher since it is fitted to the original fit rather than to the source data. The larger the deviation, the less reliable are the results for a particular choice of λ. At the optimum value

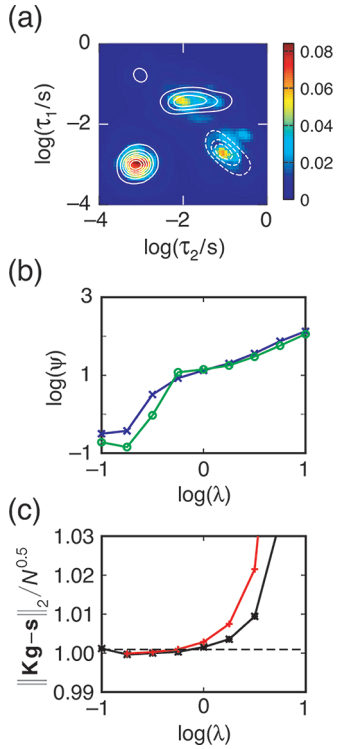

Figure 5. Self-consistency test by refitting a spectrum ten times with different random noise vectors. (a) Standard deviation of each point in the spectrum for $\lambda = 0.56$. The mean of the refitted spectrum is represented by white contour levels, with negative peaks indicated by dashed lines. The standard deviation is scaled relative to the maximum signal in the mean spectrum. (b) Comparing $\psi_{\text{norm}}(\lambda)$ (blue) and $\psi_{\text{max}}(\lambda)$ (green) for different values of λ. (c) Comparing the norm of the residuals between the source data and the original fit (black), as well as between the source data and the refit (red). The data are scaled by the square root of the inverse number of data points to facilitate an easy comparison with the noise level. The dashed horizontal line indicates the standard deviation of the noise vector. The red data point at $\lambda = 0.1$ was off the scale by several orders of magnitude and was not reproduced.

$\lambda \approx 0.5$, both the original fit and the refit show residuals that are slightly below the standard deviation of the noise added to the source data. This value of λ corresponds well with the values obtained previously by comparing $\psi_{\text{norm}}(\lambda)$ and $\psi_{\text{max}}(\lambda)$, or by using the S-curve method. This is an indirect indication that the S-curve method, which is computationally much cheaper and faster than the self-consistency test, provides a suitable λ value.

Convergence. The convergence of the algorithm is similar to the previously described 1D UP algorithm with NN penalty.[14,15] Figure 6 shows the convergence of the spectrum norm. If the regularization matrix is fully updated in each iteration of the algorithm, it is not uncommon that the ZC penalty causes an oscillation between different solutions. This applies in particular for data sets with high SNR. It can be effectively prevented by only updating the regularization matrix by a certain fraction during each iteration. Typical values, as used for all of the presented simulations, were an update ratio of initially 100%, which was decreased gradually to 20% after 30 iterations. The ZC penalty was introduced after five iterations of the algorithm and then slowly decreased to 20% after 60 iterations. As can be seen in Figure 6, the convergence rate is not strongly dependent on the SNR as long as the slope or the curvature term dominate the UP regularization matrix. For very low SNR, when the compliance floor becomes dominant, the

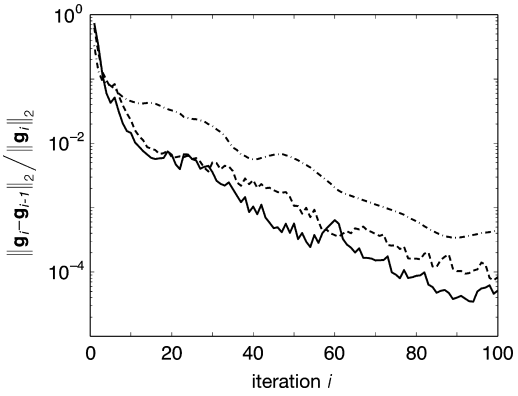

Figure 6. Convergence of the algorithm for SNR of 1000 (solid), SNR of 100 (dashed), and SNR of 10 (dash-dotted). The norm of the difference between subsequent spectra relative to the spectrum norm is plotted.

algorithm converges much more quickly. When the norm of the regularization matrix and the spectrum would not start to stabilize after about 30−50 iterations, under-regularization was usually the cause and λ had to be increased. Otherwise, the fit was terminated after 100 iterations, since the spectrum only changed marginally when more iterations were calculated.

Computation Cost. Although the presented algorithm is significantly more efficient than a brute-force implementation without compression of the source data,[14,18] it is not as fast or memory-efficient as the algorithm with an identity regularization matrix and a NN constraint.[20] On a laptop computer equipped with a 2.0 GHz Intel Celeron 550 CPU and a locally compiled version of GNU Octave 3.2.3 (http://www.octave.org/), the above calculations of a 60×60 spectrum took about 1.1 s per iteration to complete when each kernel was represented by 12 singular values. The execution time was roughly proportional to $M^{5/4}$, but this depended significantly on the CPU architecture and compilation options for Octave. The maximum spectrum size scaled approximately linearly with M. On an otherwise idle computer with 8 GB of RAM, the same system could be analyzed up to a spectrum size of $M \sim 3 \times 10^5$ elements.

The algorithm could be applied for an arbitrary number of dimensions, but on desktop computers it is only realistically possible to invert 2D and small 3D problems with the current implementation.

CONCLUSIONS

We have presented an algorithm to perform an inverse Laplace transform on multidimensional relaxation data using a generalized Tikhonov regularization. The algorithm is based on the uniform penalty regularization scheme, which was previously used successfully for 1D data, in combination with a method based on singular value decomposition to compress the source data. This algorithm does not depend on a non-negativity constraint. Such a constraint is common in current algorithms used to invert magnetic resonance relaxation data, but in the presence of cross-relaxation this constraint is physically not justified. It was shown that by using an additional penalty for the slope of the spectrum at points where the spectrum changed its sign, the dependence on the non-negativity constraint could be reduced further. Using synthetic data, the performance of the algorithm was demonstrated. It was shown that a good performance can be achieved over a large range of signal-to-noise values. On the other hand, computation costs allow for this algorithm to be used primarily for 2D and small 3D problems only.

A drawback of the presented algorithm compared to commonly used procedures for the inversion of 2D relaxation data is the large number of adjustable parameters. However, the performance of the algorithm does not critically depend on the exact choice of the different variables. On the contrary, as already reported for the 1D case,[15] UP is relatively robust in this regard, and it is usually sufficient to algorithmically select the individual α parameters and only optimize the global scaling factor λ. Developing a more fine-tuned procedure of parameter optimization will require further investigations.

Although the algorithm was only demonstrated for exponentially decaying relaxation data, its design is not restricted to this type of kernel. It should be equally applicable to other smooth kernels as well, but individual testing will be necessary. In addition, it is possible to use a unity kernel along a dimension that should not be inverted. As long as the features of the inverted dimensions are smooth along the noninverted dimension, this effects an improved SNR compared to the individual inversion of the data subsets.[31]

Eventually, by considering the results from the self-consistency test, we can attempt to phenomenologically justify the UP regularization model. If the regularization matrix is interpreted probabilistically,[7,32] then $\mathbf{\Lambda}^T\mathbf{\Lambda}$ should be equal to the inverse of the covariance matrix for spectrum $\mathbf{g}$. The UP algorithm implies that for all but the lowest SNR data, the uncertainty of the spectrum is proportional to $\mathbf{g}$, which is qualitatively confirmed by the shape of the standard deviation obtained from the self-consistency test.

AUTHOR INFORMATION

Corresponding Author

*E-mail: josef.granwehr@nottingham.ac.uk.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

We would like to thank Ana García Sanz for implementing the original non-negativity constrained algorithm[20] as part of a lab project in the School of Physics and Astronomy at the University of Nottingham. P.J.R. gratefully acknowledges a studentship from the Midlands Physics Alliance Graduate School (MPAGS).

DEDICATION

This article is dedicated to W. F. van Gunsteren in honor of his 65th birthday. J.G. would like to use this opportunity to thank him for setting up such an inspiring and spirited environment for original thinking at ETH Zurich. It was in his group that I got in contact with active research for the first time. Even as an undergraduate student, I felt welcome immediately, and I learned that research is so much more than just problem solving.

REFERENCES

(1) Istratov, A.; Vyvenko, O. *Rev. Sci. Instrum.* **1999**, *70*, 1233−1257.
(2) Williams, G.; Watts, D. *Trans. Faraday Soc.* **1970**, *66*, 80−85.
(3) Sucheta, A.; Szundi, I.; Einarsdottir, O. *Biochemistry* **1998**, *37*, 17905−17914.
(4) Gai, F.; Hasson, K.; McDonald, J. C.; Anfinrud, P. *Science* **1998**, *279*, 1886−1891.
(5) Kleinberg, R.; Kenyon, W.; Mitra, P. *J. Magn. Reson. A* **1994**, *108*, 206−214.

(6) Press, W.; Teukolsky, S.; Vetterling, W.; Flannery, B. *Numerical Recipies in C*, 2nd ed.; Cambridge University Press: Cambridge, U. K., 1992; pp 656−706.
(7) Hansen, P. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Apects of Linear Inversion*; SIAM: Philadelphia, PA, 1997; pp 69−208.
(8) Parker, R.; Song, Y.-Q. *J. Magn. Reson.* **2005**, *174*, 314−324.
(9) Provencher, S. *Comput. Phys. Commun.* **1982**, *27*, 213−227.
(10) Ernst, R.; Bodenhausen, G.; Wokaun, A. *Principles of Nuclear Magnetic Resonance in One and Two Dimensions*; Clarendon Press: Oxford, U. K., 1987; pp 490−538.
(11) Fantazzini, P.; Galassi, F.; Bortolotti, V.; Brown, R.; Vittur, F. *New J. Phys.* **2011**, *13*, 065007.
(12) Rodts, S.; Bytchenkoff, D. *J. Magn. Reson.* **2010**, *205*, 315−318.
(13) Song, Y.-Q.; Zielinski, L.; Ryu, S. *Phys. Rev. Lett.* **2008**, *100*, 248002.
(14) Borgia, G.; Brown, R.; Fantazzini, P. *J. Magn. Reson.* **1998**, *132*, 65−77.
(15) Borgia, G.; Brown, R.; Fantazzini, P. *J. Magn. Reson.* **2000**, *147*, 273−285.
(16) Barone, P.; Ramponi, A.; Sebastiani, G. *Inverse Probl.* **2001**, *17*, 77−94.
(17) Epel, B.; Pöppl, A.; Manikandan, P.; Vega, S.; Goldfarb, D. *J. Magn. Reson.* **2001**, *148*, 388−397.
(18) English, A.; Whittall, K.; Joy, M.; Henkelman, R. *Magn. Reson. Med.* **1991**, *22*, 425−434.
(19) Hürlimann, M.; Venkataramanan, L.; Flaum, C. *J. Chem. Phys.* **2002**, *117*, 10223−10232.
(20) Venkataramanan, L.; Song, Y.-Q.; Hürlimann, M. *IEEE Trans. Signal Process.* **2002**, *50*, 1017−1026.
(21) Shim, Y.; Cho, Z. *IEEE T. Acoust. Speech* **1981**, *29*, 904−909.
(22) Kiers, H. *J. Chemometr.* **2000**, *14*, 105−122.
(23) Fordham, E.; Sezginer, A.; Hall, L. *J. Magn. Reson. A* **1995**, *113*, 139−150.
(24) Chiang, Y.-W.; Borbat, P.; Freed, J. *J. Magn. Reson.* **2005**, *172*, 279−295.
(25) Butler, J.; Reeds, J.; Dawson, S. *SIAM J. Numer. Anal.* **1981**, *18*, 381−397.
(26) Golub, G.; Loan, C. V. *Matrix Computations*, 3rd ed.; The Johns Hopkins University Press: Baltimore, MD, 1996; p 50.
(27) Sato, H.; Filas, B.; Eaton, S.; Eaton, G.; Romanyukha, A.; Hayes, R.; Rossi, A. *Radiat. Meas.* **2007**, *42*, 997−1004.
(28) Granwehr, J. *Appl. Magn. Reson.* **2007**, *32*, 113−156.
(29) Callaghan, P.; Arns, C.; Galvosas, P.; Hunter, M.; Qiao, Y.; Washburn, K. *Magn. Reson. Imag.* **2007**, *25*, 441−444.
(30) Fantazzini, P.; Brown, R. *Concepts Magn. Reson. A* **2005**, *27*, 122−123.
(31) Levitt, M.; Bodenhausen, G.; Ernst, R. *J. Magn. Reson.* **1984**, *58*, 462−472.
(32) Krakauer, N.; Schneider, T.; Randerson, J.; Olsen, S. *Geophys. Res. Lett.* **2004**, *31*, L19108.