Journal of Magnetic Resonance 338 (2022) 107173

Contents lists available at ScienceDirect

Journal of Magnetic Resonance

journal homepage: www.elsevier.com/locate/jmr

Newton meets Ockham: Parameter estimation and model selection of NMR data with NMR-ESPY

Simon G. Hulse, Mohammadali Foroozandeh *

Chemistry Research Laboratory, University of Oxford, 12 Mansfield Road, Oxford, UK

ARTICLE INFO

Article history:
Received 29 November 2021
Accepted 21 February 2022
Available online 18 March 2022

Keywords:
Spectral estimation
Newton's method
Matrix Pencil
Model order selection
PYTHON
NMR-ESPY

ABSTRACT

We present NMR-ESPY (NMR Estimation in Python), a versatile, simple-to-use PYTHON package for estimating the signal parameters that describe one-dimensional time-domain NMR data. The software is fully integrated into TOPSPIN, a widely used NMR platform, and comes with a Graphical User Interface, allowing users unfamiliar with the underlying theory and/or PYTHON programming to access the full functionality of the software package. NMR-ESPY utilises Newton's method, an iterative non-linear programming technique. By including the variance of oscillator phases in the optimization, NMR-ESPY can generate parsimonious parameter estimates, giving NMR users access to meaningful quantitative information. This principle is easily extendable to study specific regions of an NMR spectrum to reduce computational cost. The complete mathematical treatment along with examples of the implementation of the estimation routine are presented.

1. Introduction

In NMR, conventional approaches to extract quantitative information consist of Fourier transformation (FT) of the detected time-domain signal (FID) [1], and inspection of the resultant Lorentzian-shaped peaks. Such a calculation is very fast on modern computer systems, even when applied to multidimensional data sets, thanks to substantial algorithmic developments [2].

Despite the simplicity of the FT approach, it suffers from two major problems: (i) the resolution limit inherent to FT, and (ii) the lack of a parsimonious representation of the data. These limitations make the task of extracting quantitative data challenging for spectra featuring resonances with similar frequencies and overlapping peaks. Such situations are common when biomolecular systems, and complex mixtures of substances are considered. Additionally, truncated FIDs lead to the appearance of "sinc wiggles" in the spectra [3, Section 3], as is often seen in solid-state NMR, and indirect dimensions of multidimensional signals. This problem can be effectively overcome by Linear Prediction algorithms [4] and non-uniform sampling (NUS) techniques [5]. However parametric estimation techniques can extend the understanding of a particular signal as they give access to spectral parameters (frequencies, amplitudes, phases, and damping factors), generally through fitting time-domain data to a model like sum of damped complex exponentials in the presence of additive stationary Gaussian noise. Such methods have found applications in many areas of science including natural language processing, audio devices, radar/sonar, and seismology [6,7].

Over the course of the past four decades, alternative methods to FT have been developed which aim to extract spectral parameters from time-domain signals. Of these, the best-known techniques are based on the separation of signal and noise sub-spaces, including Linear Prediction and Singular Value Decomposition (LPSVD) [8,9], Hankel Singular Value Decomposition (SVD) [10], and the Matrix Pencil Method (MPM) [11,12]. Approaches based on iterative nonlinear processes include the Variable Projection (VARPRO) method [13,14] and the Advanced Method for Accurate, Robust and Efficient Spectral Fitting (AMARES) [15]. More recently, methods drawn on Bayesian analysis have also emerged [16–18]. Whilst these techniques offer many benefits over the classical FT approach, they are typically more computationally costly. To address this problem, techniques relying on the idea of sub-band decomposition, initially illustrated with Linear Prediction-ZOOM (LP-ZOOM) [19], have been developed. For further information and discussions of more recent developments see [20,21] and references therein.

One of the main reasons that many of the proposed methods have not found widespread application, and techniques beyond peak picking and integration in the frequency-domain have struggled to enter the mainstream in NMR is that the role of spectral estimation is generally underestimated, especially among NMR

* Corresponding author.
E-mail address: mohammadali.foroozandeh@chem.ox.ac.uk (M. Foroozandeh).
URL: http://foroozandeh.chem.ox.ac.uk/home (M. Foroozandeh).

https://doi.org/10.1016/j.jmr.2022.107173
1090-7807/

experimentalists who consider any non-linear post-processing method with scepticism. Another main contributor is that usually, these techniques are not very straightforward to use, as often the user is required to be familiar with the theory and programming to use the method effectively.

In this work we present NMR-ESPY (NMR Estimation in Python) to address these problems. NMR-ESPY is a free Python package, for both accurate and parsimonious estimation of NMR spectral parameters. It comes with a powerful Graphical User Interface (GUI) that provides NMR users a seamless interface for benefiting from the full functionalities of NMR-ESPY without the need to write a single line of code. Additionally the whole software and GUI are fully integrated into Bruker's TopSpin software and are accessible from the command line.

The core functionality of NMR-ESPY relies of effective use of regularised non-linear optimisation. It is known that a non-linear iterative techniques, such as Newton's method [22,23], can give access to accurate estimation of spectral parameters. However, such a technique typically requires substantial *a priori* knowledge, in the form of an initial parameter estimate, in order to function effectively, especially when a large number of signal components are under consideration. In this regard, an initial guess is generated using the Information-Theoretic Matrix Pencil Method (ITMPM), pioneered by Pines and co-workers [24], based on the MPM of Hua and Sarkar [11]. Another critical aspect of the method is the prediction of the number of signal modes (oscillators), as well as their phase behaviour. This is achieved by an implementation of a routine that minimises not only the discrepancy between the data and model, but also the variance of oscillator phases. This technique is shown to be effective at purging excessive oscillators from an initial parameter guess, avoiding over-estimation (overfitting), and leading to a more reliable and parsimonious representation of the data. In order to enhance the computational efficiency of the method, a filtration scheme using virtual echo signals [25,26] is utilised, enabling estimation of spectral sub-bands with both fewer oscillators and data points, resulting in drastically reduced computational cost.

This paper is structured as follows: in Section 2 the theoretical aspects of the method are discussed in detail; in Section 3 two examples of its application on NMR data are presented, and finally, in Section 4, an overview of the NMR-ESPY package and the GUI is outlined, presenting its main features.

1.1. Nomenclature

In this text, vectors and matrices are denoted by lower-case bold letters, $\boldsymbol{v}$, and upper-case bold letters, $\boldsymbol{M}$, respectively. Some of the operators used are as follows:

z^* - complex conjugate of a scalar z
$|z|$ - absolute value of $z := \sqrt{z^* z}$
$\lfloor z \rfloor$ - the closest integer to z, satisfying $\lfloor z \rfloor \leqslant z$
$\lceil z \rceil$ - the closest integer to z, satisfying $\lceil z \rceil \geqslant z$
$\boldsymbol{v}^{\mathrm{T}}$ - transpose of $\boldsymbol{v}$
$\boldsymbol{v}^\dagger$ ($\boldsymbol{M}^\dagger$) - Hermitian transpose of $\boldsymbol{v}$ ($\boldsymbol{M}$)
$\|\boldsymbol{v}\|_2$ - Euclidean (ℓ^2) norm of $\boldsymbol{v} := \sqrt{\boldsymbol{v}^\dagger \boldsymbol{v}}$
$\boldsymbol{v}_1 \odot \boldsymbol{v}_2$ - element-wise (Hadamard) product of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$
$\boldsymbol{M}^{-1}$ - inverse of square nonsingular matrix $\boldsymbol{M}$
$\boldsymbol{M}^+$ - Moore–Penrose pseudoinverse of $\boldsymbol{M}$

The following notations are used for sets:

$\mathbb{R}$ - the set of real numbers
$\mathbb{R}_{>0}$ - the set of greater than 0 real numbers
$\mathbb{C}$ - the set of complex numbers
$\mathbb{N}_0$ - the set of natural numbers, with 0 included

2. Theory

2.1. Outline of the Problem

In NMR, a discrete 1D time-domain dataset (FID), $\boldsymbol{y} = [y_0, y_1, \cdots, y_{N-1}]^{\mathrm{T}} \in \mathbb{C}^N$, is assumed to comprise two main components: (i) a non-random, modellable contribution $\boldsymbol{x}(\boldsymbol{\theta}) = [x_0, x_1, \cdots, x_{N-1}]^{\mathrm{T}}$, dependent on the free parameter vector $\boldsymbol{\theta}$, which can be defined as a sum of M damped complex exponentials (modes), and (ii) a random noise component $\boldsymbol{w} = [w_0, w_1, \cdots, w_{N-1}]^{\mathrm{T}}$, generally considered to be white, Gaussian noise.

$$\boldsymbol{y} = \boldsymbol{x}(\boldsymbol{\theta}) + \boldsymbol{w}, \tag{1a}$$

$$x_n(\boldsymbol{\theta}) = \sum_{m=1}^{M} a_m \exp(\mathrm{i}\phi_m) \exp[(2\pi \mathrm{i} f_m - \eta_m)\tau_n], \tag{1b}$$

$$\boldsymbol{\theta} = [a_1, \cdots, a_M, \phi_1, \cdots, \phi_M, f_1, \cdots, f_M, \eta_1, \cdots, \eta_M]^{\mathrm{T}}, \tag{1c}$$

with $n \in \{0, 1, \cdots, N-1\}$. $a_m := \theta_m \in \mathbb{R}_{>0}$, $\phi_m := \theta_{m+M} \in (-\pi, \pi]$, $f_m := \theta_{m+2M} \in [-f_{\mathrm{sw}}/2 + f_{\mathrm{off}}, f_{\mathrm{sw}}/2 + f_{\mathrm{off}}]$ and $\eta_m := \theta_{m+3M} \in \mathbb{R}_{>0}$ denote the amplitude, phase, frequency and damping factor of oscillator m, respectively. f_{sw} is the spectral width (the inverse of the dwell time, Δt), and f_{off} is the transmitter frequency offset. τ_n is the $(n+1)$th time step at which the data was sampled. For uniformly-sampled data (the focus of this paper), τ_n is equivalent to $n\Delta t$.

In more succinct notation, the model expression for the FID is given by

$$x_n = \sum_{m=1}^{M} \alpha_m z_m^n, \tag{2}$$

where α_m and z_m are the *complex amplitude* and the *signal pole* of the m^{th} signal component, respectively, and can be written as

$$\alpha_m = a_m \exp(\mathrm{i}\phi_m), \tag{3a}$$

$$z_m = \exp[(2\pi \mathrm{i} f_m - \eta_m)\Delta t], \tag{3b}$$

$\forall m \in \{1, 2, \cdots, M\}$. The full model vector $\boldsymbol{x}$ may therefore be expressed in a vector and matrix form as $\boldsymbol{x} = \boldsymbol{Z}\boldsymbol{\alpha}$, where $\boldsymbol{Z} \in \mathbb{C}^{N \times M}$ is a Vandermonde matrix containing the signal poles, and $\boldsymbol{\alpha} \in \mathbb{C}^M$ is a vector containing the complex amplitudes. s

$$\begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_M \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_M^{N-1} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_M \end{pmatrix}. \tag{4}$$

The noise elements w_n are assumed to be complex and have independent real and imaginary components, with $\mathrm{Re}(w_n)$, $\mathrm{Im}(w_n) \sim \mathcal{N}(0, \sigma^2) \Rightarrow w_n \sim \mathscr{C}\mathcal{N}(0, 2\sigma^2)$. The probability density function (pdf) for an individual noise component is

$$p(w_n) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{|w_n|^2}{2\sigma^2}\right). \tag{5}$$

As the noise elements are independent and identically distributed (iid), the joint pdf describing the noise vector $\boldsymbol{w}$ is given by the product of all of its constituents' pdfs:

$$\begin{aligned} p(\boldsymbol{w}) &= \prod_{n=0}^{N-1} \tfrac{1}{2\pi\sigma^2} \exp\left(-\tfrac{|w_n|^2}{2\sigma^2}\right) \\ &= \tfrac{1}{(2\pi\sigma^2)^N} \exp\left(-\tfrac{\|\boldsymbol{w}\|_2^2}{2\sigma^2}\right). \end{aligned} \tag{6}$$

Since the noise vector is simply the difference between the data and the model vectors (Eq. (1a)), the likelihood function $\mathscr{L}(\boldsymbol{\theta} \mid \boldsymbol{y})$ is

$$\mathscr{L}(\boldsymbol{\theta} \mid \boldsymbol{y}) = \frac{1}{(2\pi\sigma^2)^N} \exp\left(-\frac{1}{2\sigma^2}\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{\theta})\|_2^2\right), \tag{7}$$

and log-likelihood $\ell := \ln(\mathscr{L})$ is therefore

$$\ell(\boldsymbol{\theta} \mid \boldsymbol{y}) = -N\ln\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{\theta})\|_2^2. \tag{8}$$

As the application of the natural logarithm is a monotonic transformation, Eq. (8) implies that an optimal parameter estimation of the model can be obtained by choosing a vector $\hat{\boldsymbol{\theta}}$ that minimises $\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{\theta})\|_2^2$:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^{4M}} -\ell(\boldsymbol{\theta} \mid \boldsymbol{y}) \equiv \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^{4M}} \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{\theta})\|_2^2. \tag{9}$$

To solve such a problem, a non-linear programming (NLP) routine, e.g. Newton's method, seems an ideal choice. However, iterative procedures of this type typically require a reasonably good initial guess $\boldsymbol{\theta}^{(0)}$ in order to achieve a guaranteed convergence to a local minimum of the cost function, which is especially pertinent when dealing with high-dimensional problems [7]. Another important issue is that the number of oscillators, M, which dictates the dimension of $\boldsymbol{\theta}$, is generally unknown. While one solution to this issue may be to apply FT to $\boldsymbol{y}$ and inspect the number of visible peaks in the frequency-domain data, it is unlikely to be feasible for complicated spectra featuring heavily overlapping resonances, covering a high dynamic range, and with low SNR. These cases are where spectral estimation is most desirable. Consequently, before using Newton's method to estimate $\hat{\boldsymbol{\theta}}$, it is necessary to produce an initial estimate which, ideally, requires no *a priori* knowledge about the FID. Application of one of the numerous eigenanalysis methods mentioned in the introduction is an attractive means of achieving this. In this work, we have opted to generate $\boldsymbol{\theta}^{(0)}$ using the ITMPM [24], discussed in Section 2.2, which combines detection theory [27] with estimation theory [11].

Remark 1. In this work, all FIDs under consideration were normalised prior to any manipulation or estimation. As such, a data-set $\boldsymbol{y}$, is actually $\boldsymbol{y}/\|\boldsymbol{y}\|_2$.

2.2. Generating an Initial Guess Using the ITMPM

The concept of model order selection is based on Ockham's razor; the best signal model is the one with the fewest necessary independent parameters. If the predicted model order M is too small, information present in the data will not be extracted. Conversely, if M is excessive, unwanted effects, such as the modelling of individual resonances by multiple oscillators, and incorporation of noise components in the model (i.e. overfitting) will occur.

The conventional route to model order selection involves consideration of the log-likelihood, $\ell(\boldsymbol{\theta} \mid \boldsymbol{y})$. It was shown by Wax and Kailath [27] that $\ell(\boldsymbol{\theta} \mid \boldsymbol{y})$ can be represented as

$$\ell(\boldsymbol{\theta} \mid \boldsymbol{y}) = \ln\left(\frac{\prod_{r=k+1}^{L}\lambda_r^{1/(L-k)}}{\frac{1}{L-k}\sum_{r=k+1}^{L}\lambda_r}\right)^{(L-k)N}, \tag{10}$$

where λ_r is the r^{th} largest eigenvalue of the sample covariance matrix. These eigenvalues relate directly to the singular values σ_r of the Hankel matrix $\boldsymbol{Y}$

$$\boldsymbol{Y} = \begin{pmatrix} y_0 & y_1 & \cdots & y_L \\ y_1 & y_2 & \cdots & y_{L+1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N-L-1} & y_{N-L} & \cdots & y_{N-1} \end{pmatrix}. \tag{11}$$

L is commonly referred to as the pencil parameter, which can be seen to dictate the number of rows and columns $\boldsymbol{Y}$ is composed of. It was subsequently shown that $\ell(\boldsymbol{\theta} \mid \boldsymbol{y})$ in Eq. (10) can be combined with a penalising function, $\mathscr{P}(k)$, to form a cost function $\mathscr{C}(k)$,

$$\mathscr{C}(k) = c\ell(\boldsymbol{\theta} \mid \boldsymbol{y}) + \mathscr{P}(k), \tag{12}$$

where c is a constant. The model order can be predicted via minimisation of $\mathscr{C}(k)$. Two of the most well-known criteria for model order selection are the Akaike Information Criterion (AIC) [28] and the Minimum Description Length (MDL) [29,30]:

$$\mathrm{AIC}(k) = -2\ell(\boldsymbol{\theta} \mid \boldsymbol{y}) + 2k(2L-k), \tag{13a}$$

$$\mathrm{MDL}(k) = -\ell(\boldsymbol{\theta} \mid \boldsymbol{y}) + \frac{1}{2}k(2L-k)\ln N, \tag{13b}$$

$\forall k \in \{0, 1, \cdots, L-1\}$. Beyond the AIC and MDL, there are other ways to estimate model order, including the Maximum *a posteriori* (MAP) probability [31], and Penalising Adaptively the Likelihood (PAL) [32]. For our purposes, the MDL has been employed, such that the model order M is estimated by solving

$$M = \arg\min_{k} \mathrm{MDL}(k). \tag{14}$$

The Matrix Pencil Method (MPM) is an SVD-based technique for extracting the signal poles contained within a signal. In brief, given an estimated number of oscillators M, one may generate an low-rank approximation of the signal matrix $\boldsymbol{Y}$ using its M most significant singular vectors and values:

$$\widetilde{\boldsymbol{Y}} = \widetilde{\boldsymbol{U}}\widetilde{\boldsymbol{\Sigma}}\widetilde{\boldsymbol{V}}^{\dagger}, \tag{15a}$$

$$\widetilde{\boldsymbol{U}} = [\,\boldsymbol{u}_1 \quad \boldsymbol{u}_2 \quad \cdots \quad \boldsymbol{u}_M\,], \tag{15b}$$

$$\widetilde{\boldsymbol{V}} = [\,\boldsymbol{v}_1 \quad \boldsymbol{v}_2 \quad \cdots \quad \boldsymbol{v}_M\,], \tag{15c}$$

$$\widetilde{\boldsymbol{\Sigma}} = \mathrm{diag}(\sigma_1, \sigma_2, \cdots, \sigma_M), \tag{15d}$$

where $\boldsymbol{u}_m \in \mathbb{C}^{N-L}$ and $\boldsymbol{v}_m \in \mathbb{C}^{L+1}$ correspond to the left and right singular vectors associated with the m^{th} largest singular value σ_m. The signal poles $\{z_m^{\mathrm{MP}}\}, m \in \{1, 2, \cdots, M\}$ are then determined via computation of the eigenvalues of the matrix $\widetilde{\boldsymbol{V}}_1^{+}\widetilde{\boldsymbol{V}}_2$, with $\widetilde{\boldsymbol{V}}_1 = [\,\boldsymbol{v}_1 \quad \boldsymbol{v}_2 \quad \cdots \quad \boldsymbol{v}_{M-1}\,]$ and $\widetilde{\boldsymbol{V}}_2 = [\,\boldsymbol{v}_2 \quad \boldsymbol{v}_3 \quad \cdots \quad \boldsymbol{v}_M\,]$ obtained via removal of the first/last column of $\widetilde{\boldsymbol{V}}$, respectively. For a more detailed review of the principles underpinning the MPM, see Section B in the Supplementary Information, and references [11,24].

Once the poles have been estimated using the MPM, due to their linear dependence, the complex amplitudes α_m^{MP}, can be derived by solving the linear least-squares problem

$$\begin{aligned} \boldsymbol{\alpha}^{\mathrm{MP}} &= \arg\min_{\boldsymbol{\alpha}\in\mathbb{C}^M} \left\|\boldsymbol{Z}^{\mathrm{MP}}\boldsymbol{\alpha} - \boldsymbol{y}\right\|_2^2, \\ &\Rightarrow \boldsymbol{\alpha}^{\mathrm{MP}} = \left(\boldsymbol{Z}^{\mathrm{MP}}\right)^{+}\boldsymbol{y}, \end{aligned} \tag{16}$$

where according to Eq. (4)

$$\boldsymbol{\alpha}^{\mathrm{MP}} = \left[\,\alpha_1^{\mathrm{MP}} \quad \alpha_2^{\mathrm{MP}} \quad \cdots \quad \alpha_M^{\mathrm{MP}}\,\right]^{\mathrm{T}}, \tag{17a}$$

$$\boldsymbol{Z}^{\mathrm{MP}} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ z_1^{\mathrm{MP}} & z_2^{\mathrm{MP}} & \cdots & z_M^{\mathrm{MP}} \\ \vdots & \vdots & \ddots & \vdots \\ \left(z_1^{\mathrm{MP}}\right)^{N-1} & \left(z_2^{\mathrm{MP}}\right)^{N-1} & \cdots & \left(z_M^{\mathrm{MP}}\right)^{N-1} \end{pmatrix}. \tag{17b}$$

The amplitudes, phases, frequencies, and damping factors corresponding to each oscillator, which make up the free parameters of

the vector $\boldsymbol{\theta}^{\mathrm{MP}} \in \mathbb{R}^{4M}$ are then determined using the following relations:

$$\theta_m^{\mathrm{MP}} \equiv a_m^{\mathrm{MP}} = \|\boldsymbol{y}\|_2 \left|\alpha_m^{\mathrm{MP}}\right|, \tag{18a}$$

$$\theta_{m+M}^{\mathrm{MP}} \equiv \phi_m^{\mathrm{MP}} = \arctan\left(\frac{\mathrm{Im}\left(\alpha_m^{\mathrm{MP}}\right)}{\mathrm{Re}\left(\alpha_m^{\mathrm{MP}}\right)}\right), \tag{18b}$$

$$\theta_{m+2M}^{\mathrm{MP}} \equiv f_m^{\mathrm{MP}} = \frac{f_{\mathrm{sw}}}{2\pi}\mathrm{Im}\left(\ln z_m^{\mathrm{MP}}\right) + f_{\mathrm{off}}, \tag{18c}$$

$$\theta_{m+3M}^{\mathrm{MP}} \equiv \eta_m^{\mathrm{MP}} = -f_{\mathrm{sw}}\mathrm{Re}\left(\ln z_m^{\mathrm{MP}}\right), \tag{18d}$$

Note that the presence of $\|\boldsymbol{y}\|_2$ in Eq. (18a) reflects the fact that the data is normalised for analysis (see *Remark 1*).

Notable downsides of the MPM include the computational cost, and memory requirements. The accuracy and resolution of the MPM is reliant on the value of the pencil parameter L, the most effective value being $\lfloor N/2 \rfloor$. However, larger values of L and N drastically increase the cost of computing the SVD of $\widetilde{\boldsymbol{Y}}$, with complexity $\mathcal{O}\left(\min\left((N-L)^2L, (N-L)L^2\right)\right)$. NMR FIDs often comprise 16k (2^{14}) or more data points, and for these signals - assuming that the computer used has the memory capacity to perform the SVD in the first place - the amount of time to generate the signal poles can make the MPM an unattractive method. There has been recent work in the pursuit of improving the efficiency of the MPM using localised alternatives [33]. For our purposes, we only desire an initial guess for non-linear programming. As such, with large speed-ups, $\boldsymbol{\theta}^{\mathrm{MP}}$ is produced using a sub-optimal pencil parameter ($\lfloor N/3 \rfloor$ has been used in all work presented in this paper). Often, if appropriate, it is also possible to analyse a truncated signal with minimal loss of estimation accuracy, further reducing the computational burden.

2.3. Nonlinear Programming Using Newton's Method

Having obtained an initial estimate of the spectral parameters using model order selection and the MPM, we next employ a non-linear programming methodology. As discussed in Section 2.1, a common approach to estimate the optimal parameter vector $\hat{\boldsymbol{\theta}}$ which reproduces $\boldsymbol{y}$ is to minimise the fidelity $\mathscr{F}(\boldsymbol{\theta}) : \mathbb{C}^N \times \mathbb{R}^{4M} \to \mathbb{R}$ with $\mathscr{F}(\boldsymbol{\theta}) = \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{\theta})\|_2^2$, With Newton's method, one approximates the fidelity as quadratic about the current iterate, $\boldsymbol{\theta}^{(k)}$ ($k \in \mathbb{N}_0$):

$$\mathscr{F}(\boldsymbol{\theta}) \approx \mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right) + \boldsymbol{h}^{\mathrm{T}}\nabla\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right) + \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}\nabla^2\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right)\boldsymbol{h}, \tag{19}$$

where $\boldsymbol{h} = \boldsymbol{\theta} - \boldsymbol{\theta}^{(k)}$, $\nabla\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right) \in \mathbb{R}^{4M}$ is the gradient of $\mathscr{F}$, and $\nabla^2\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right) \in \mathbb{R}^{4M\times 4M}$ is its Hessian, elements of which are

$$\left[\nabla\mathscr{F}\right]_i = -2\mathrm{Re}\left[(\boldsymbol{y}-\boldsymbol{x})^{\dagger}\frac{\partial\boldsymbol{x}}{\partial\theta_i}\right], \tag{20a}$$

$$\left[\nabla^2\mathscr{F}\right]_{ij} = -2\mathrm{Re}\left[(\boldsymbol{y}-\boldsymbol{x})^{\dagger}\frac{\partial^2\boldsymbol{x}}{\partial\theta_i\partial\theta_j} - \frac{\partial\boldsymbol{x}^{\dagger}}{\partial\theta_i}\frac{\partial\boldsymbol{x}}{\partial\theta_j}\right], \tag{20b}$$

where θ_i (θ_j) is the i^{th} (j^{th}) element of $\boldsymbol{\theta}$, with $i,j \in \{1,2,\cdots,4M\}$. Due to the independent nature of the oscillators, all first derivatives of the model vector $\boldsymbol{x}$ can be written using 4 general expressions, and all non-zero model vector second derivatives can be written using 10 general expressions (Section C in the Supplementary Information). The general principle of Newton's method is that the succeeding iterate $\boldsymbol{\theta}^{(k+1)}$ is determined to be the minimum of the quadratic approximation, such that

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - s^{(k)}p^{(k)}, \tag{21a}$$

$$p^{(k)} = \left[\nabla^2\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right)\right]^{-1}\nabla\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right), \tag{21b}$$

where $s^{(k)}$ is the step length, and $p^{(k)}$ is the Newton direction. It should be noted that in practice there are numerous algorithmic variants that exploit this concept, which differ in their details [22,23]. The process of determining improved estimates is repeated until a convergence criterion are met, typically involving consideration of the gradient norm $\left\|\nabla\mathscr{F}\left(\boldsymbol{\theta}^{(k)}\right)\right\|_2$ with the final result $\hat{\boldsymbol{\theta}}$ being the parameter estimate. The vector of the standard error of the estimation, $\mathrm{SE}\left(\hat{\boldsymbol{\theta}}\right) \in \mathbb{R}^{4M}$, can be calculated using the expression

$$\mathrm{SE}\left(\hat{\boldsymbol{\theta}}\right) = \sqrt{\frac{\mathscr{F}\left(\hat{\boldsymbol{\theta}}\right)\mathrm{diag}\left(\left[\nabla^2\mathscr{F}\left(\hat{\boldsymbol{\theta}}\right)\right]^{-1}\right)}{N-1}}, \tag{22}$$

a derivation of which is presented in Section D in the Supplementary Information.

2.4. Invoking Ockham's Razor: Minimisation of Phase Variance

Although Newton's method can return a parameter estimate $\hat{\boldsymbol{\theta}}$ that results in a highly accurate reconstruction of the original NMR data, the apparent goodness of fit can hide a subtle but serious problem: over-estimation of the model order, i.e. the number of oscillators. One upshot of this is that a single resonance in the data may be modelled using several oscillators with similar frequencies but spurious phases, which significantly degrades the meaningfulness of the estimation, as desired properties such as peak integrals and chemical shifts become poorly defined.

To overcome this problem, a modified fidelity, $\mathscr{F}^{\phi}$, is utilised in which the estimation accuracy is not only quantified in terms of the discrepancy between the data and model, but simultaneously in terms of the variance of phases.

$$\mathscr{F}^{\phi}(\boldsymbol{\theta}) = \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{\theta})\|_2^2 + \frac{1}{\pi}\mathrm{Var}(\boldsymbol{\phi}), \tag{23a}$$

$$\mathrm{Var}(\boldsymbol{\phi}) = \frac{1}{M}\sum_{m=1}^{M}\left(\phi_m - \mu_{\phi}\right)^2. \tag{23b}$$

where $\boldsymbol{\phi} := [\theta_{M+1}, \theta_{M+2}, \cdots, \theta_{2M}]^{\mathrm{T}} = [\phi_1, \phi_2, \cdots, \phi_M]^{\mathrm{T}}$, and μ_{ϕ} is the mean of the elements in $\boldsymbol{\phi}$. The first and second partial derivatives of the phase variance are given in Section C.2 in the Supplementary Information.

Remark 2. It should be noted that if the data is not normalised during the optimisation routine, as data acquired on NMR spectrometers typically spans many orders of magnitude, the phase variance would have a negligible impact on $\mathscr{F}^{\phi}$.

Minimisation of $\mathscr{F}^{\phi}$ is obtained using an unconstrained NLP procedure (i.e. $\boldsymbol{\theta}$ is not subjected to any restrictions). Of course, it is not physically reasonable for the parameters to acquire any value within $\mathbb{R}$ (the restrictions on the parameters are stated in 2.1). Therefore, as the optimisation proceeds, if any oscillator within $\boldsymbol{\theta}$ acquires a negative amplitude, it is purged from the model. We demonstrate in Section 3 that this technique can be effective at removing excessive oscillators, leading to a more parsimonious representation of the data.

2.5. Estimation of a Specified Spectral Region

The bottleneck of this NLP routine is the computation of the Hessian matrix, which scaled quadratically with the number of model oscillators and linearly with the number of points (see Section 2.6). As such, the number of oscillators that feature in the model will have a significant impact on estimation time. A drastic reduction in computation time can be realised when only a particular region of a spectrum is of interest, by application of band-pass frequency-filtration (Fig. 1), as this will reduce the model order M.

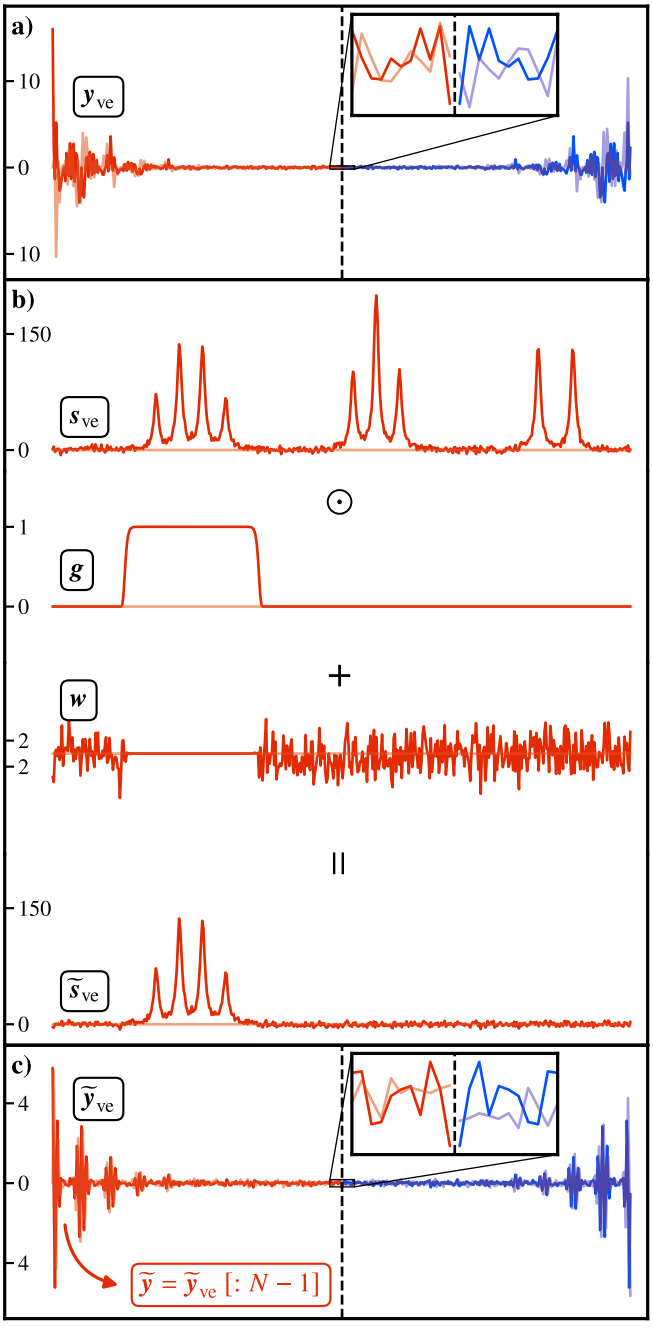


Fig. 1. a) A virtual echo generated from an FID comprising N points. The conjugate symmetry of the signal is illustrated by the zoomed-in region in the middle of the signal. **b)** The Fourier transform of the virtual echo leads to a spectrum with zero imaginary component. This is filtered via application of a super-Gaussian filter with the addition of synthetic noise. **c)** The filtered frequency-domain data is the transferred back to the time domain, generating a filtered virtual echo, of which, the first $N-1$ points are considered in estimation.

The initial step in the filtration process is to construct a *virtual echo* [25,26] $\boldsymbol{y}_{ve} \in \mathbb{C}^{2N-1}$ from $\boldsymbol{y}$:

$$\boldsymbol{y}_{ve} = \left[(y_0 + y_0^*)/2, y_1, \cdots, y_{N-1}, y_{N-1}^*, y_{N-2}^*, \cdots, y_1^* \right]^T. \tag{23c}$$

$\boldsymbol{y}_{ve}$ can be considered as the concatenation of the following two vectors:

(i) $\boldsymbol{y}$, with the only real component of the first point retained.
(ii) $\boldsymbol{z} \in \mathbb{C}^{N-1}$, with elements defined by $z_n = y_{N-1-n}^* \; \forall n \in \{0, 1, \cdots, N-2\}$

The Fourier transform of conjugate symmetric $\boldsymbol{y}_{ve}$ generates a spectrum $\boldsymbol{s}_{ve}$ with zeros for the imaginary component. Phasing $\boldsymbol{s}_{ve}$ such that its real part comprises absorption Lorentzian peaks produces a frequency-domain signal with no dispersion contribution, a property that is desirable as it minimises the frequency-span required to adequately select a particular resonance. The frequency subspace of interest is then selected by the application of a band-pass filter, for example, using an appropriate super-Gaussian function $\boldsymbol{g}$

$$g_n = \exp\left(-2^{p+1}\left(\frac{n-c}{b}\right)^p\right), \quad n \in \{0, 1, \cdots, 2N-2\}, \tag{23d}$$

where b is the number of points the filter spans (its bandwidth), and c is the index of the centre of the filter. A value of $p = 40$ was used for all the analysis presented here, and for the example filter presented in Fig. 1.b), which provides a smooth transition at the cut-off frequencies, and a constant amplitude throughout the selected region. Since the super-Gaussian filter will turn regions outside the specified window to 0, and thus remove any noise associated with those regions, any attempt at model order selection will result in a significant over-estimation of M. Therefore, a vector of synthetic noise $\boldsymbol{w}$, with elements $w_n \sim \mathcal{N}(0, \sigma_n^2)$ is added to the filtered spectrum, where the associated variance at a particular point n is dependent on the amplitude of the super-Gaussian at that point, i.e. $\sigma_n^2 = (1 - g_n)\sigma^2$, where σ^2 is the estimated noise variance of the spectrum, given by the variance of a region in the spectrum where no discernible peaks reside.

Inverse Fourier Transformation (IFT) of the filtered spectral data leads to the formation of another virtual echo, $\tilde{\boldsymbol{y}}_{ve}$, where the tilde denotes filtering. This is finally sliced, with only the first $N-1$ points retained, creating a frequency filtered signal $\tilde{\boldsymbol{y}}$:

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{y}}_{ve}[0 : N-1], \tag{23e}$$

where

$$\tilde{\boldsymbol{y}}_{ve} = \mathrm{IFT}\left\{\mathrm{FT}^{\phi}\{\boldsymbol{y}_{ve}\} \odot \boldsymbol{g} + \boldsymbol{w}\right\}. \tag{23f}$$

$[\alpha : \beta]$ is a slicing notation, indicating that values starting at index α and ending at index $\beta - 1$ are retained, and FT^{ϕ} denotes Fourier transformation along with phasing.

Application of this frequency filtration leads to a small subset of the frequency space within which signal information is contained. Therefore, it is appropriate to truncate the frequency domain signal prior to IFT ($\tilde{\boldsymbol{s}}_{ve,t} \in \mathbb{C}^{<2N-1}$), by removing spectral regions that are sufficiently far away from the region of interest, without any loss of information. To achieve this, a region is defined, centred about c, with a bandwidth of γb, where $\gamma > 1$ is a scaling factor. In the examples presented in this paper, $\gamma = 3$. See Section E of the Supplementary Information for more details.

2.6. Hessian Computation Time

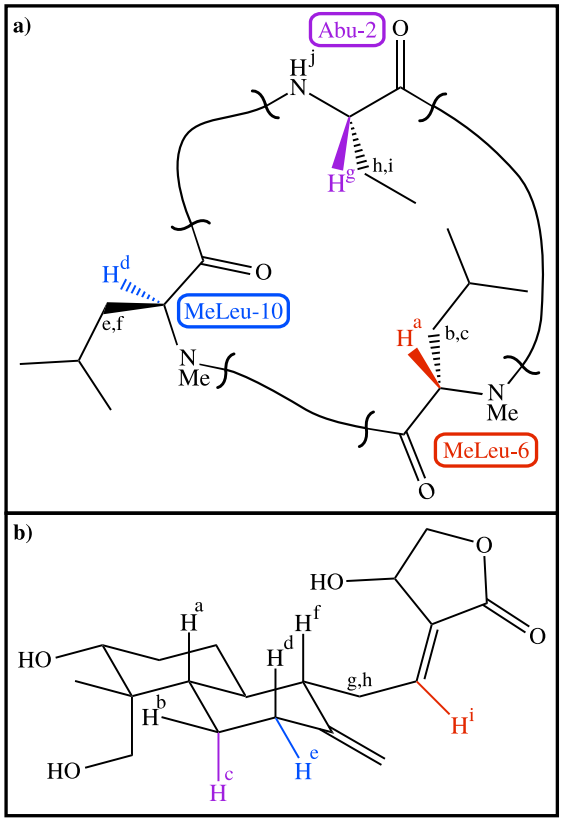
To gain insight into the computational cost of generating the Hessian matrix $\nabla^2\mathscr{F}(\boldsymbol{\theta})$, a series of such calculations were run for signals with varying numbers of points, N, and oscillators, M. For each $N \in \{512l \mid l \in \{1,2,\cdots,16\}\}$, synthetic data were constructed with $M \in \{4k \mid k \in \{1,2,\cdots,16\}\}$ using a parameter vector $\hat{\boldsymbol{\theta}} \in \mathbb{R}^{4M}$. The SNR of each synthetic data-set was set to 20 dB by addition of white Gaussian noise. The parameter vector $\hat{\boldsymbol{\theta}}$ was then used to compute the Hessian matrix for each data-set and the wall-clock time was recorded. The procedure was repeated five times for each pair of M and N, to obtain an average wall-clock time (Fig. 2).

Remark 3. Note that the Hessian matrix needs to be inverted, which typically has a complexity $\mathcal{O}\left((4M)^3\right)$ (for algorithms involving Gaussian elimination at least). However, for this particular problem, the time to compute the inverse is orders of magnitude faster than the computation of the Hessian matrix.

Note that the Hessian matrix can be computed in $\mathcal{O}\left(M^2N\right)$ time, i.e. it scales linearly with the number of points in the FID, and quadratically with the number of modes in the model. This explains the significant reduction in the computational burden when the proposed band-pass frequency filtration and signal truncation is used. Less computationally expensive alternatives to the Newton method, in which an analytic Hessian is required, are quasi-Newton methods, including BFGS and DFP [22,23], in which the Hessian matrix is approximated using the cost function and analytic gradients, albeit at the cost of super-linear instead of quadratic convergence.

3. Examples

In this section, two examples of the application of the proposed estimation technique are presented. All results were obtained using the NMR-ESPY package, on a workstation with an Intel® Core™-i9-10900X CPU @ 3.70 GHz, and 32GiB of RAM. The optimisation routine used in all cases was `scipy.optimize.minimize()`, with the method set to a Trust Region algorithm (`'trust-constr'`).



Scheme 1. The structures of the species whose ^{1}H signals are considered in this text. (**a**) A simplified representation of cyclosporin A, with only the amino acids possessing protons of interest denoted. (**b**) The structure of andrographolide. The proton environments giving rise to the multiplet patterns considered are coloured, and match the peaks in Figs. 3 and 4.

Scheme 1 shows a simplified representation of cyclosporin A and the structure of andrographolide, colour-coded for the protons giving rise to the estimated resonances.

3.1. Cyclosporin Data

Fig. 3 shows applications of our estimation routine on two regions of a ^{1}H spectrum of a sample of 50 mM cyclosporin A

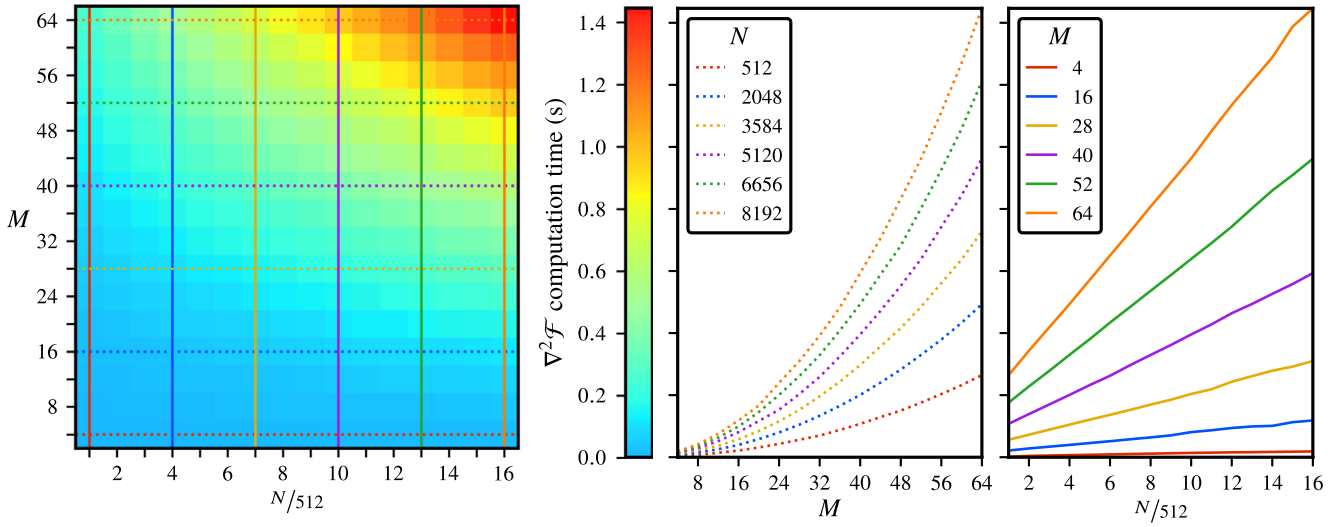


Fig. 2. The dependence of Hessian computation time on the number of data points (N) and the number of oscillators (M). From left to right: A heat map indicating the computation time as a function of M and N; the variation of computation time as a function of M, for six distinct values of N; the variation of computation time as a function of N, for six distinct values of M.

(a cyclic non-ribosomal peptide comprising 11 amino acids) in benzene-d_6. The two regions selected were 5.540–5.420 ppm and 5.287–5.185 ppm, regions for in which it is common to locate peptide H^{α} nuclei. The former region contains resonances arising from two N-methylleucine moieties (MeLeu-6, MeLeu-10), and the latter contains resonances arising from 2-aminobutyric acid (Abu-2) [34]. The structures of the three residues of interest are depicted in Scheme 1a. For both regions studied, an initial guess $\boldsymbol{\theta}^{(0)}$ was generated via model selection using the MDL, followed by the MPM. $\boldsymbol{\theta}^{(0)}$ was then subjected to Newton's method with the inclusion of a phase variance penalty (Eq. (23b)), producing the final result $\hat{\boldsymbol{\theta}}$.

It can be seen in Fig. 3a that the MPM has been effective at producing parameter estimates which closely represent the data. However, in both regions considered, there are low-amplitude oscillators residing close to the boundaries of the region of interest (1 & 12 and 1 & 11 in the two regions, respectively). It is likely that these were generated as a side-effect of the super-Gaussian filtration procedure. As well as this, in the 5.540–5.420 ppm region result, another spurious low intensity oscillator (6) with a phase close to $\pi/2$ exists. All of the mentioned oscillators were automatically purged during the phase variance-penalised Newton method. In the 5.287–5.185 ppm region, it can be seen that the MPM was able to detect two low-amplitude resonances, indicated by † and ‡, which have been largely hidden by the intense multiplet arising from H^{α}(Abu-2). Despite this, the estimated phases were unexpected, given the well-phased nature of the data. The final result $\hat{\boldsymbol{\theta}}$ in Fig. 3b shows a far better agreement with the expected phase behaviour.

3.2. Andrographolide Data

Fig. 4 shows applications of our estimation routine on three regions of a ^{1}H spectrum of a sample of 40 mM andrographolide in DMSO-d_6. The three regions feature multiplets from the protons indicated in Scheme 1b. In similar fashion to the cyclosporin examples, in all three cases the number of oscillators predicted by the MDL was greater than the number after application of the phase variance-penalised Newton method, as a result of purging superfluous oscillators, leading to parsimonious estimation results. A full

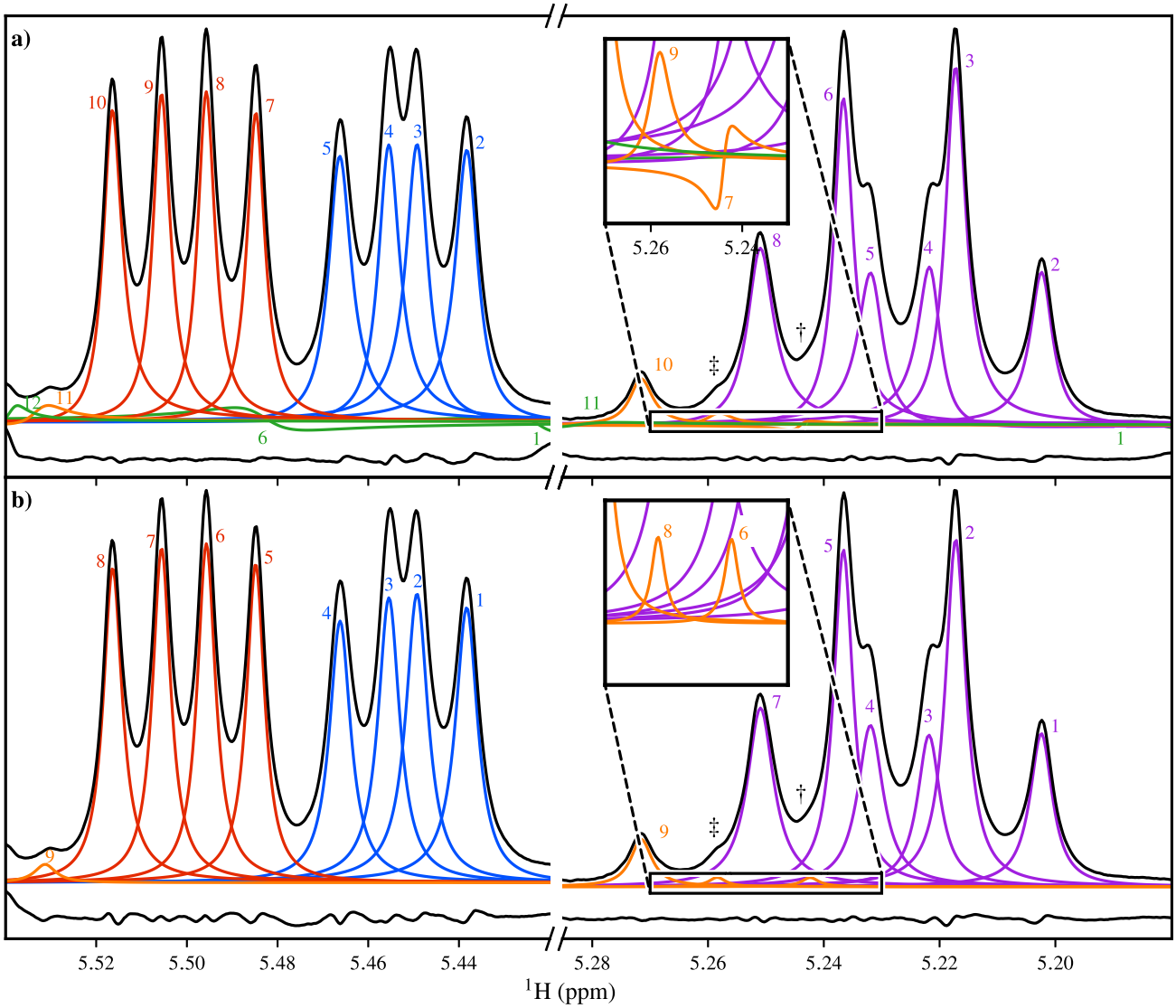


Fig. 3. An illustration of the application of the estimation routine to two regions of a ^{1}H spectrum of 50 mM cyclosporin A in benzene-d_6. Spectra of the regions of interest, 5.54–5.42ppm and 5.287–5.185 ppm, respectively, are plotted alongside the results from **a)** the MPM, with model order selection using the MDL **b)** the phase variance-penalised Newton method with the result from **(a)** as the initial guess. Also, the residual of the data and the summation of all estimated peaks is plotted in each panel. Note that peaks are colour-coded to match the corresponding proton environments given in Fig. 1a. Orange peaks denote oscillators in the model which do not come from the protons of interest, likely from impurities. Green peaks are for oscillators which exist in the MPM result, but are purged during .the NLP routine.

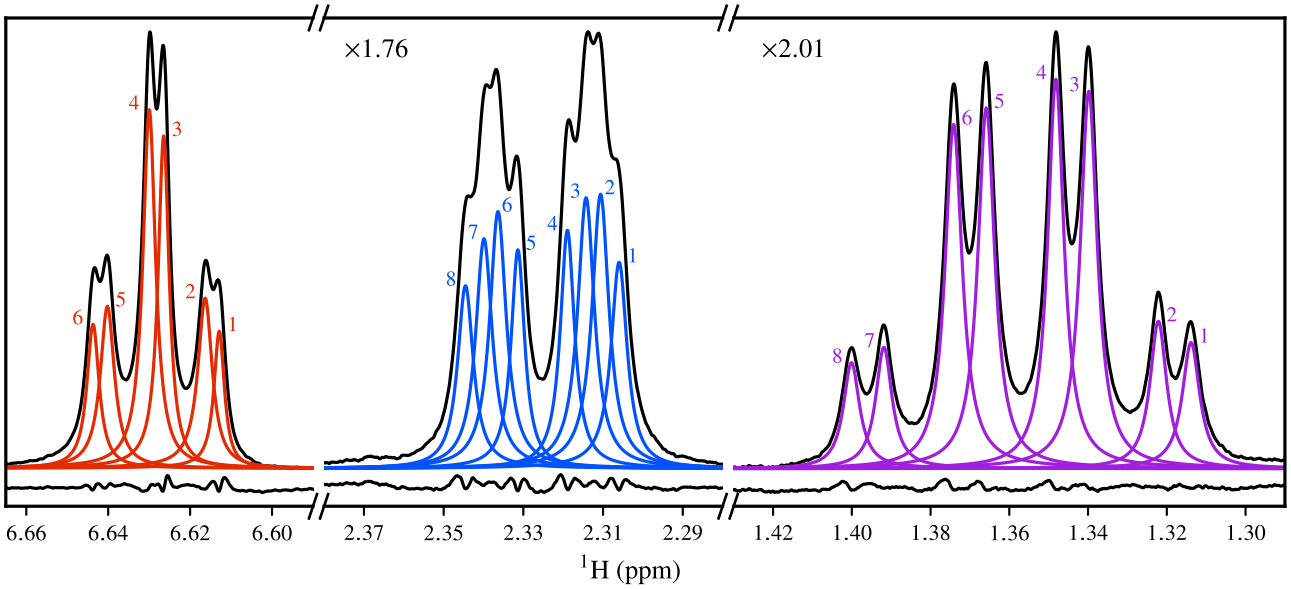


Fig. 4. The results of running the estimation routine on three regions of a ^{1}H NMR spectrum of andrographolide. Each oscillator in the final result $\hat{\boldsymbol{\theta}}$ is plotted along with the residual between the sum of oscillators and the data.

assignment of andrographolide can be found in [35, Page 340]. Partial assignments for multiplet structures of cyclosporin A and andrographolide studied here are summarised in Section F in the Supplementary Information. Tables of complete results, including estimation errors, are presented in Section G in the Supplementary Information.

4. The NMR-ESPY Package

The NMR-ESPY package processes NMR data via extensive use of NumPy and SciPy [36,37]. At the time of writing, NMR-ESPY supports the direct import of data stored in Bruker format. We plan to provide functionality for data from other NMR manufacturers including VARIAN and JEOL, in the future.

The typical workflow for using NMR-ESPY is as follows:

1. Import the data and relevant parameters, such as sweep width and transmitter offset. Both raw FID data (stored in a `fid` file for Bruker systems), and processed data (stored in `lr`) are supported.
2. Optional, but recommended: Generate a frequency filtered signal corresponding to the user-defined spectral region of interest.
3. Produce an initial guess using the Matrix Pencil Method. It is possible to either estimate the number of oscillators using the MDL, or for the user to specify the number of desired oscillators manually.
4. Run an NLP routine to generate the final estimation result. The user can choose whether or not to include the phase variance penalty in the fidelity, which may or may not be suitable based on the dataset being analysed.
5. Output the result to various file formats (`.txt`, `.pdf`, and `.csv` are supported at the time of writing). Figures of the estimation result can also be generated in an automated fashion.

Remark 4. The estimation routine relies on the assumption that the time-domain signal analysed possesses oscillators that are exponentially damped. This translates to Lorentzian line shapes in the Fourier domain. Whilst processed data that has been zero-filled and baseline corrected should still produce time-domain signals that are valid under this assumption, certain window functions such as Lorentz-Gaussian and sine bell will render the data incompatible with the model. Although, in general, in the case of deviation from Lorentzian lineshape, one can retrieve Lorentzian lineshapes, for example, via reference deconvolution methods [38], before the application of the estimation routine.

All these steps are straightforward to carry out, due to a simple-to-use high-level API which NMR-ESPY provides. As an example of using NMR-EsPy, the script in Fig. 5 was used to generate the

```
#!/usr/bin/python3
from nmrespy.core import Estimator

# Create estimator instance
path = "path/to/.../1/pdata/1"
estimator = Estimator.new_bruker(path)

# Generate ỹ_t
estimator.frequency_filter(
    region = ((5.287, 5.185),),
    noise_region = ((6.48, 6.38),),
)

# Generate θ^(0)
estimator.matrix_pencil()
# Generate θ̂, by minimising 𝓕^ϕ
estimator.nonlinear_programming(phase_variance=True)

# Save result to .txt, .pdf and .csv formats
desc = "50mM cyclosporin in Benzene-d6"
for fmt in ("txt", "pdf", "csv"):
    estimator.write_result(
        path="result",
        fmt=fmt,
        description=desc,
    )

# Create a figure of the estimation result
plot = estimator.plot_result()
plot.fig.savefig("result_figure.pdf")
```

Fig. 5. An example of a typical script highlighting the basic workflow for utilising NMR-ESPY.

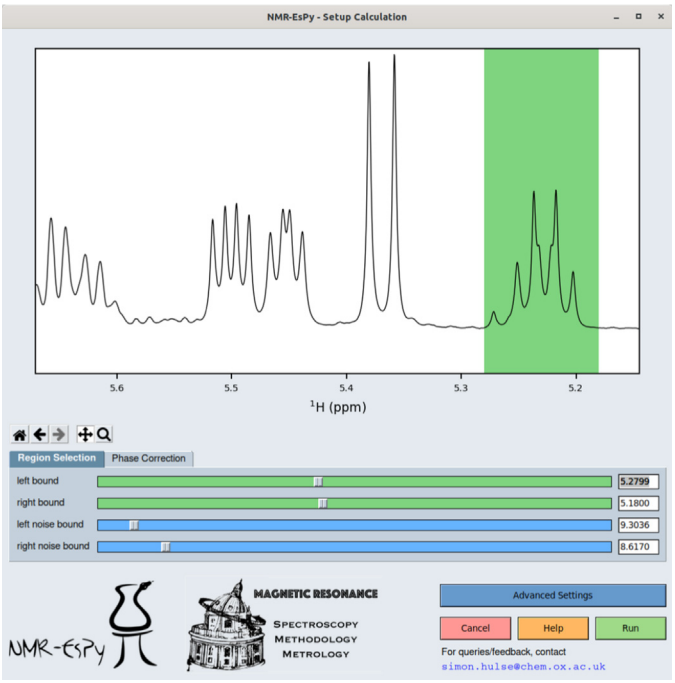


Fig. 6. A screenshot of the NMR-ESPY GUI on a system with Ubuntu 20.04. The green region in the plot indicates the region to estimate. The GUI also provides the user a simple means of phasing the data, and to customise various aspects of the routine via the 'Advanced Settings' button.

estimation results presented in the right side of Fig. 3. NMR-ESPY also has a Graphical User Interface (GUI) which can be loaded either by entering `nmrespy` into Bruker's TOPSPIN command prompt, or from a terminal (Fig. 6). The GUI gives access to all of NMR-ESPY's features without the requirement to write a single line of code.

For more information about the NMR-ESPY package, the reader is referred to:

- The source code, hosted on GitHub at https://github.com/foroozandehgroup/NMR-EsPy.
- Sections H (code snippets illustrating how the Matrix Pencil Method and optimisation routines are implemented) and I (links to documentation) in the Supplementary Information.

We plan to include various additional features to the package in the future, including support for two-dimensional data, and welcome any bug reports or suggestions for additional features.

5. Conclusions

In the present work, we have introduced a new PYTHON package, NMR-ESPY, for spectral estimation and optimisation of 1D NMR data, giving access to the accurate spectral parameters (frequencies, amplitudes, phases, and damping factors). NMR-ESPY is fully integrated into Bruker's TopSpin software and has a GUI. We have demonstrated that the combination of spectral estimation methods, e.g. Matrix Pencil method with Non-Linear Programming in the form of Newton's method with a phase variance-penalised fidelity can lead to a more reliable and parsimonious representation of NMR data. We believe NMR-ESPY can significantly facilitate the process of spectral analysis and bring a major benefit to the field of NMR with applications including the study of biomolecules, metabolomics, complex mixtures, pharmaceuticals, and reaction monitoring.

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.

Acknowledgements

This work was funded by the Royal Society (Grant Nos. URF|R1|180233 and RGF|EA|181018). We wish to thank Timothy Claridge for helpful discussions, and Jonathan Yong for supplying the andrographolide dataset.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.jmr.2022.107173.

References

[1] R.R. Ernst, Nuclear magnetic resonance fourier transform spectroscopy (nobel lecture), Angew. Chem. Int. Ed. 31 (7) (1992) 805–823.
[2] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex fourier series, Math. Comput. 19 (90) (1965) 297–301.
[3] J. Lindon, A. Ferrige, Digitisation and data processing in fourier transform nmr, Prog. Nucl. Magn. Reson. Spectrosc. 14 (1) (1980) 27–66.
[4] P. Koehl, Linear prediction spectral analysis of nmr data, Prog. Nucl. Magn. Reson. Spectrosc. 34 (3–4) (1999) 257–299.
[5] M. Mobli, J.C. Hoch, Nonuniform sampling and non-fourier signal processing methods in multidimensional nmr, Prog. Nucl. Magn. Reson. Spectrosc. 83 (2014) 21–41.
[6] S.M. Kay, Fundamentals of statistical signal processing: estimation theory, international ed Edition, Prentice-Hall signal processing series, Prentice-Hall International, London, 1993.
[7] P. Stoica, R.L. Moses, Spectral analysis of signals, Pearson Education, Upper Saddle River, NJ, 2005.
[8] R. Kumaresan, D. Tufts, Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise, IEEE Trans. Acoust. Speech Signal Process. 30 (6) (1982) 833–840.
[9] H. Barkhuijsen, R. de Beer, W. Bovée, D. van Ormondt, Retrieval of frequencies, amplitudes, damping factors, and phases from time-domain signals using a linear least-squares procedure, J. Magn. Reson. 61 (3) (1985) 465–481.
[10] H. Barkhuijsen, R. de Beer, D. van Ormondt, Improved algorithm for noniterative time-domain model fitting to exponentially damped magnetic resonance signals, J. Magn. Reson. 73 (3) (1987) 553–557.
[11] Y. Hua, T. Sarkar, Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise, IEEE Trans. on Acoust. Speech Signal Process. 38 (5) (1990) 814–824.
[12] Y. Hua, T. Sarkar, On svd for estimating generalized eigenvalues of singular matrix pencil in noise, IEEE Trans. Signal Process. 39 (4) (1991) 892–900.
[13] G.H. Golub, V. Pereyra, The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate, SIAM J. Numer. Anal. 10 (2) (1973) 413–432.
[14] J.W. van Der Veen, R. de Beer, P.R. Luyten, D. van Ormondt, Accurate quantification of in vivo 31p nmr signals using the variable projection method and prior knowledge, Magn. Reson. Med. 6 (1) (1988) 92.
[15] L. Vanhamme, A. van Den Boogaart, S. Van Huffel, Improved method for accurate and efficient quantification of mrs data with use of prior knowledge, J. Magn. Reson. 129 (1) (1997) 35–43.
[16] D.V. Rubtsov, J.L. Griffin, Time-domain bayesian detection and estimation of noisy damped sinusoidal signals applied to nmr spectroscopy, J. Magn. Reson. 188 (2) (2007) 367–379.
[17] D.V. Rubtsov, C. Waterman, R.A. Currie, C. Waterfield, J.D. Salazar, J. Wright, J.L. Griffin, Application of a bayesian deconvolution approach for high-resolution (1)h nmr spectra to assessing the metabolic effects of acute phenobarbital exposure in liver tissue, Anal. Chem. 82 (11) (2010) 4479.
[18] K. Krishnamurthy, Craft (complete reduction to amplitude frequency table) – robust and time-efficient bayesian approach for quantitative mixture analysis by nmr, Magn. Reson. Chem. 51 (12) (2013) 821–829.
[19] J. Tang, J. Norris, Lp-zoom, a linear prediction method for local spectral analysis of nmr signals, J. Magn. Reson. 79 (1) (1988) 190–196.
[20] N. Sandgren, Y. Selén, P. Stoica, J. Li, Parametric methods for frequency-selective mr spectroscopy–a review, J. Magn. Reson. 168 (2) (2004) 259–272.
[21] D.E.-H, T. M, B. D, Nmr data analysis: A time-domain parametric approach using adaptive subband decomposition, Oil Gas Sci. Technol. 69 (2) (2014) 229–244.
[22] R. Fletcher, Practical methods of optimization, 2nd Edition., Wiley, Chichester, 1987.
[23] J. Nocedal, S.J. Wright, Numerical optimization, 2nd Edition, Springer series in operations research, Springer, New York, 2006.

[24] Y.-Y. Lin, P. Hodgkinson, M. Ernst, A. Pines, A novel detection–estimation scheme for noisy nmr signals: Applications to delayed acquisition data, J. Magn. Reson. 128 (1) (1997) 30–41.
[25] M. Mayzel, K. Kazimierczuk, V.Y. Orekhov, The causality principle in the reconstruction of sparse nmr spectra, Chem. Commun. 50 (64) (2014) 8947–8950.
[26] D. Golowicz, P. Kasprzak, K. Kazimierczuk, Enhancing compression level for more efficient compressed sensing and other lessons from nmr spectroscopy, Sensors (Basel) 20 (5) (2020) 1325.
[27] M. Wax, T. Kailath, Detection of signals by information theoretic criteria, IEEE Trans. Acoust. Speech Signal Process. 33 (2) (1985) 387–392.
[28] H. Akaike, A new look at the statistical model identification, IEEE Trans. Automat. Control 19 (6) (1974) 716–723.
[29] G. Schwarz, Estimating the dimension of a model, Ann. Stat. 6 (2) (1978) 461–464.
[30] J. Rissanen, Modeling by shortest data description, Automatica 14 (5) (1978) 465–471.
[31] P. Djuric, Asymptotic map criteria for model selection, IEEE Trans. Signal Process. 46 (10) (1998) 2726–2735.
[32] P. Stoica, P. Babu, Model order estimation via penalizing adaptively the likelihood (pal), Signal Process. 93 (11) (2013) 2865–2871.
[33] E. Aboutanios, D.S. Thomas, J.M. Hook, C. Cobas, Locmap: A new localization method for the parametric processing of high resolution nmr data, J. Magn. Reson. 282 (2017) 62–70.
[34] A. Verma, S. Bhattacharya, B. Baishya, Perfecting band selective homodecoupling for decoupling two signals coupled within the same band, RSC Adv. 8 (2018) 19990–19999.
[35] T.D.W. Claridge, High-resolution NMR techniques in organic chemistry, third edition. Edition, Amsterdam, 2016.
[36] C.R. Harris, K.J. Millman, S.J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, Array programming with numpy. (report), Nature 585 (7825) (2020) 357.
[37] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C.J. Carey, İ. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nat. Methods 17 (2020) 261–272.
[38] G.A. Morris, H. Barjat, T.J. Home, Reference deconvolution methods, Prog. Nucl. Magn. Reson. Spectrosc. 31 (2) (1997) 197–257.