Deep regression with ensembles enables fast, first-order shimming in low-field NMR

Moritz Becker, Mazin Jouda, Anastasiya Kolchinskaya, Jan G. Korvink *

Karlsruhe Institute of Technology (KIT), Institute of Microstructure Technology, Karlsruhe 76131, Germany

Abstract

Shimming in the context of nuclear magnetic resonance aims to achieve a uniform magnetic field distribution, as perfect as possible, and is crucial for useful spectroscopy and imaging. Currently, shimming precedes most acquisition procedures in the laboratory, and this mostly semi-automatic procedure often needs to be repeated, which can be cumbersome and time-consuming. The paper investigates the feasibility of completely automating and accelerating the shimming procedure by applying deep learning (DL). We show that DL can relate measured spectral shape to shim current specifications and thus rapidly predict three shim currents simultaneously, given only four input spectra. Due to the lack of accessible data for developing shimming algorithms, we also introduce a database that served as our DL training set, and allows inference of changes to 1H NMR signals depending on shim offsets. In situ experiments of deep regression with ensembles demonstrate a high success rate in spectral quality improvement of random shim distortions over different neural architectures and chemical substances. This paper presents a proof-of-concept that machine learning can simplify and accelerate the shimming problem, either as a stand-alone method, or in combination with traditional shimming methods. Our database and code are publicly available.

© 2022 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent decades, deep learning (DL) [1] has shown unprecedented achievements in various industrial and scientific fields. Although areas of research such as computer vision or natural language processing have become commonplace, many fields of science have only recently begun to exploit the vast possibilities that DL can provide. One such area is nuclear magnetic resonance (NMR) spectroscopy, a non-destructive technique widely used in chemistry, physics and medicine to study the properties of liquid or solid samples. The most notable contributions from DL currently employed in NMR include, but are not limited to, methods for reconstruction of non-uniformly sampled (NUS) spectra [2] and truncated free induction decays (FIDs) [3], chemical shift prediction [4], or denoising and segmentation of magnetic resonance images [5,6]. Other possible applications of DL for solving challenges in NMR are also discussed and suggested in the community [7].

Most of these approaches and suggestions are being applied at the post-processing stages of NMR spectroscopy, taking the hardware setup for the NMR measurement as granted. In contrast, we propose to optimize the preparation preceding an NMR measurement with deep learning methods.

One crucial parameter of the NMR setup, which requires the most careful and precise tuning for subsequent successful measurements, is the homogeneity of the magnetic field. The strength of the magnetic field $B_0$ directly influences the precession frequency $f$ of each stationary spin with its gyromagnetic ratio $\gamma$, as described by the well-known Larmor equation $2\pi f = -\gamma B_0$. For a perfectly uniform field and a chemically homogeneous sample, the observable signal in the frequency domain yields a single Lorentzian peak centered at $f$. However, in an inhomogeneous magnetic field $B_0$, the spins experience different local fields due to various effects (e.g., susceptibility differences between material inside the field, manufacturing inaccuracies of the coil, or intramolecular shielding effects in the sample itself), and thus possess frequencies shifted around the central Larmor frequency. This means that inhomogeneities broaden the line shapes of the spectrum, decrease their amplitude and consequently decrease the signal-to-noise ratio (SNR). Furthermore, the non-bijective mapping of the FID from a three-dimensional volume to a one-dimensional spectrum introduces ambiguities, i.e., the connection between the location of each distortion and its direct impact on
the spectrum is lost during the acquisition process (Fig. 1). Especially with recent efforts in miniaturizing NMR technologies [8], signal sensitivity increases, but field inhomogeneities remain hard to eliminate.

To overcome distortions in the measured spectrum caused by a magnetic field inhomogeneity, a method referred to as “shimming” is used. Modern shimming is best described as the procedure of superimposing a secondary correction magnetic field by adjusting the currents in a finite set of field-orthogonal coils (the so-called shim coils) to correct for inhomogeneities in the magnetic field $B_0$. Modern NMR spectrometers usually require the magnetic field to be uniform in the range of parts per billion (ppb).

However, the shimming procedure is often a tedious and painstaking process that demands extensive time and experience from the operator [9]. This is due to the large number of shim coils required for spectroscopy, the inter-dependence of shim field patterns (violations of orthogonality), and the lack of a straightforward solution for the correct shim coil values in an (expected) non-convex solution space. This situation makes it a challenge to provide a correct, fast, and reliable shimming procedure for the magnetic field, especially on the order of a few seconds.

Several approaches already exist to solve this problem and automate the shimming procedure to increase spectral quality. The most robust approaches to date utilize the downhill simplex (or Nelder-Mead) method [10,11], and adaptations thereof [12]. Also, automated shimming based on a lock channel is widely used, e.g. for continuously adapted shimming during lengthy NMR measurements. Unfortunately, with large initial field inhomogeneities, locking may be impossible due to its inherently low SNR. A significant contribution to shimming utilizes gradient shimming [13,14]. For this, ideally, rapidly switchable gradient coils would be necessary, which may not be available. Also, the $B_0$ field maps could be distorted [15], requiring appropriate corrections and potentially repetitions. Despite emerging solutions [16–18], signal-based methods remain significant and can automatically improve spectral quality if sufficient runtime is provided. Nevertheless, the achieved improvements are often not perfect, and manual refinement is necessary due to imperfect hardware, non-orthogonal or dependent shims [19], and sensitivity to starting values and step sizes for regular shimming methods.

We propose to fill this gap by utilizing a deep learning (DL) algorithm to guide the shimming process. As a tool, DL has demonstrated great success over various domains by end-to-end learning, i.e., an algorithm learns, given only input and target, to automatically detect features. In general, the feature detection is enabled by combining representations in multiple layers based on representations from the previous layer, where each layer represent more abstract feature levels [1]. With sufficient capacity, DL can thus learn arbitrary complex functions of high-dimensional input, and still allows for fast inference. Therefore, we hypothesize that it is possible for DL to learn shim values given 1D signals drawn from 3D space, even when the shim values do not directly correspond to visible features in the signal. Accordingly, we utilize supervised deep regression due to its capability to automatically detect non-linear relations between high-dimensional input data and numerical targets with high performance. We furthermore merge deep regression with the idea of ensembles, by combining multiple weak models to reduce prediction variance.

We focus on a non-iterative method for initial shimming to rapidly reach a state near the global optimum. Our scenario assumes shimming a probe from scratch with first-order shims, and beneficial use cases are the acceleration or improvement of existing automated shimming methods, focusing on high-throughput NMR alone or in conjunction with miniaturized hardware [20], without the use of gradients or even a lock channel. For this, we have generated a publicly available database for first-order NMR shimming that allows inference of spectral changes depending on shim offsets. We utilize the database to train a set of deep regression models (so-called weak learners) that can simultaneously predict three first-order shim currents given four distinct NMR measurements: the current unshimmed spectrum, and three spectra with individually modified shim values. The weak learners are then combined in an ensemble, via a meta-model, to increase prediction stability, and the performance is analyzed in situ for different evaluation metrics. Furthermore, we conduct limited comparison with regular shimming based on the downhill simplex method.

In summary, our paper makes the following contributions:

- Creation of the first spectral database (ShimDB) dedicated to low-field NMR shimming¹;
- Proof-of-concept for utilizing deep learning for shimming;
- Establishment of a method for rapid shimming based on deep regression with ensembles (DRE)¹, shown in Fig. 2.

Note that we only use information learned from data, i.e., we employ a knowledge-based approach. We also do not require priors, or mathematical formulations of spatial shim functions, as would be required for gradient shimming.

The rest of the paper is organized as follows. Section 2 provides an overview of related work on automated shimming, deep regression, and ensembles. In Section 3, we introduce our database, and in Section 4, our method. In Section 5, we describe our experiments for both DL training and in situ deployment. We discuss our approach in Section 6 and conclude in Section 7.

2. Fundamentals and related work

We start by unfolding the shimming problem and highlighting some of the successful or recent approaches. We also describe relevant advances in deep learning that we adopted in our method.

---

2.1. Automated shimming

The shimming problem. The nuclear spin ensemble precession frequencies differ depending on the locally experienced and spatially varying field strengths of an inhomogeneous magnetic field $B_0$ at a 3D coordinate $r = (r_1, r_2, r_3)$. Mathematically, the shimming algorithm finds the scalar weights $w_i$ for $n$ shim currents such that the corrected magnetic field $B_0$ is as uniform as possible:

$$
\overline{B_0}(r) = B_0(r) + \sum_{i=1}^{n} w_i S_i(r).
$$

The basis set of (ideally orthogonal) spatial functions $S_i, i \leq n \in \mathbb{N}$, and their real scalar weights $w_i$ are adjusted to reproduce and thus cancel the field inhomogeneities $\Delta B_0$. Each $S_i$ represents a specific shim coil (or spatial shim coil function) with its current $w_i$. An example of the shimming procedure for an arbitrary axis in space is given in Fig. 3. Magnetic field inhomogeneities cause distortions in the spectrum, which can be canceled by sequentially adding spatial shim functions $S_i$ with the correct weights $w_i$. The concept of homogenization by superimposing correction fields was developed by Golay in 1958 [22].

In general, according to their optimization objective, automated shimming methods can be separated into signal-based and field-based methods. Field-based methods, or gradient shimming [13], use $B_0$ field maps acquired by gradient-echo imaging sequences to calculate an optimal combination of basis functions to cancel inhomogeneities. We focus on iterative signal-based shimming, which optimizes a scalar quality criterion based on signals such as the FID, lock channel, or spectral lineshapes. Moreover, we can neglect theoretical limitations of spatial functions and directly optimize for the shim currents.

1-D signal-based shimming. These are based on 1-D search algorithms, covered by the Tuning [23] or Coggins [24] algorithms, and are characterized by optimizing one variable at a time. The methods all share the procedure of repeatedly comparing three spectra until the minimum quality criterion of choice can be approximated by fitting a parameterized parabolic curve. Adjusting one shim at a time is simple and often faster than other methods, but does not incorporate dependencies between shim values. This is why it often has to be iterated.

N-D signal-based shimming. In n-dimensional optimization, a group of $n$ variables are adjusted simultaneously. For this, the downhill simplex method [11], or its modifications [12], are usually used. The methods commonly use a geometrical polytope (a "simplex") of $n+1$ vertexes, where each vertex is represented by the quality criterion corresponding to specific shim settings. The simplex then evolves through solution space using geometrical
operations such as reflection, expansion, and contraction of the simplex, based on the worst, average and best quality criteria, until a local minimum is reached, evidenced by vertices of similar quality criterion. Note that the Nelder-Mead simplex method [10] applicable in shimming should not be confused with the simplex algorithm of Dantzig for linear programming [25].

The major limitation of the downhill simplex method is its slow convergence speed [26]. Several improvements, such as quasi-gradient methods [27], adaptive shrinking coefficients [28] or perturbed centroids [29], are combined in [12] to enhance NMR shimming.

Other methods, such as modified steepest descent [11], or the rapid, modified simplex method proposed by Webb et al. [30], also enable n-dimensional shim optimization.

Other approaches. The main idea behind the method introduced by Michal [19] is to orthogonalize the shim coil gradients such that the optimization of one shim becomes independent of the others. With truly-orthogonal “composite shims”, finding a global minimum becomes a one-dimensional calculation. The method brings various limitations: gradients show non-linear behaviour due to current supply, heating, or other components, and thus affect the symmetry required to calculate composite shims.

2.2. Deep learning methodologies

We now consider deep learning methods, a branch of machine learning, which learns representations from a set of prior data by abstraction at multiple levels.

Deep learning overview. Deep neural networks are most commonly implemented as stacked layers of artificial neurons, where the weight of each neuron is updated (or learned) with backpropagation [31] and gradient descent, to minimize a loss function on given training data. However, generalisation is judged by the prediction performance on previously unseen (out-of-sample) data. Therefore, different regularization techniques can prevent overfitting or unwanted memorization of the training samples. For example, early stopping of the training process is used when the training and test errors diverge, dropout [32] randomly shuts down neurons during weight update, and augmentation is used to increase the amount and variance of data. The usage of non-linear activation functions, such as the rectified linear unit (ReLU) [33], with a sufficient number of processing layers (depth), allows neural networks to approximate any arbitrary function. Furthermore, different connection patterns, e.g. each-to-each (fully-connected), of the neurons in and between each layer, allow for efficient prediction on different data structures (e.g. sequential or image-like). We refer to [1] for a more detailed description of the entire DL idea.

Deep Regression. The generalization of standard regression (i.e. prediction of a single continuous variable) to multiple, possibly interdependent variables is often referred to as multi-target regression [34]. When deep neural networks replace the function approximation, one commonly refers to deep regression. In [35], vanilla deep regression is coined to refer to convolutional neural networks (CNNs) with a linear last layer. Spectral reconstruction and denoising methods [5] in NMR can be seen as regression functions, such as the rectified linear unit (ReLU) [33], with a sufficient number of processing layers (depth), allowing neural networks to approximate any arbitrary function. Furthermore, different connection patterns, e.g. each-to-each (fully-connected), of the neurons in and between each layer, allow for efficient prediction on different data structures (e.g. sequential or image-like). We refer to [1] for a more detailed description of the entire DL idea.

3. First-order shimming dataset

The success of deep learning algorithms strongly depends on the quantity and quality of available data, which should represent the task as accurate as possible without introducing biases. Thus, the first step in the development of DL-assisted automatic shimming was the creation of a database for algorithm training. The shimming database (ShimDB)2 is a collection of proton NMR signals recorded under the application of shim coil fields and so far contains a subset called LinearShimDB, a small-scale dataset containing over 9000 instances with only linear shim offsets. It allows inference of changes to the NMR spectrum or free induction decay (FID) depending on linear shim offsets. Each data instance includes the following information: A binary file containing the raw 1H-FID with dimensions 1 x 32768; the shim values \( \epsilon \) for n shims; the acquisition parameters; and the processing parameters. We pretend that the spectrometer only has n = 3 first-order shims, thereby only X, Y, and Z shim values are non-zero, but this is easily extended.

The measured sample consists of distilled water mixed with copper sulfate to reduce spin–lattice or longitudinal relaxation time \( T_1 \), allowing for faster database acquisition. In analogy to the study [48], the 50 ml H2O is mixed with 0.062 g CuSO4·5H2O (CAS No. 7758–99–8), resulting in a concentration of 5 mMol/L. With inversion recovery experiments we find that \( T_1 = 290 \text{ ms} \).

The data was acquired on the low-field Magritek Spincontrol 80 Carbon spectrometer (Magritek GmbH, Aachen Germany, [49]) with a 1H frequency of 80 MHz using standard 5 mm sample tubes and the Spinparse-Expert software. Experimental parameters, and the dataset’s characteristics, are summarized in Table 1. A large reception bandwidth was chosen because the initial line shape is unknown when using large shim offsets. Also, the frequency lock is not activated so that the signal potentially could leave the field of view.

---

2 It is intended to continuously extend the database with new subsets.
The acquisition procedure of the LinearShimDB subset was as follows. The manufacturer’s automated shimming technique, based on the downhill simplex method [50], was used to obtain a reference spectrum of decent quality. Then, all shim values except the three linear shims \(X, Y, Z\) were set to a current of zero Ampère. The resulting spectrum and corresponding shim settings \(y\) were used as the reference values. The database parameters were obtained by relative, systematic offsets and shim values were stored.3

We believe that our data can successfully be reused for training ML models on different setups and scenarios by transfer learning or domain adaptation techniques [51].

4. Deep regression with ensembles for shimming

In this section, we define the mathematical problem and introduce our network structures for deep regression, where we differentiate between weak learners (level-0), and the meta-model (level-1). Furthermore, we introduce our performance metrics.

4.1. Problem definition in terms of DL and NMR

Let \(\mathcal{D} = \{(\mathbf{x}, \mathbf{y})\}_{i=1}^{D}\) be our database, where \((\mathbf{x}, \mathbf{y})\), is a pair including the input \(\mathbf{x} \in \mathbb{R}^{W \times 4}\) with dimensions \(W \times 4\), where \(W\) is the input’s width, and the associated target \(\mathbf{y} = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n\), defined as a real-valued vector of \(n\) elements, with \(n\) being the number of separate shim coils. Also, consider the regression model \(\mathcal{F}_\theta(\cdot)\), represented by a deep convolutional neural network with parameters \(\theta\). The network parameters \(\theta\) are learned in a supervised manner using the database \(\mathcal{D}\) in order to minimize the mean squared error (MSE) between the prediction \(\mathbf{y} = \mathcal{F}_\theta(\mathbf{x})\) and the target \(\mathbf{y}\).

In terms of NMR, the predictions \(\mathbf{y}\) translate/map to the shim values \(w\), and should solve Eq. 1. The inputs \(\mathbf{x}\) are defined as \(\mathbf{x} = (u(0), u(y_1), u(y_2), u(y_3))\), where the unshimmmed spectrum \(u\) is defined as a function of systematic shim offsets \(s\). The DL model predicts the shim correction terms \(\mathcal{F}_\theta(\mathbf{x}) = (\hat{y}_1, \hat{y}_2, \hat{y}_3)\), such that \(\mathbf{y} - \mathbf{y}_0 = 0\). Note that we do not have access to either \(S_0\), or the magnetic field \(B_0\). We also compare the mean absolute error (MAE) to measure in situ shimming performance between the predicted and relative offset values of the reference shims (see Section 3) with \(\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |\mathbf{y}_i - \hat{\mathbf{y}}_i|\).

4.2. Deep learning architectures and pipeline

Level-0. Unlike in computer vision, in NMR there is no objective mapping between the input to the model and its output (i.e. prediction at the pixel level of the input). Indeed, the shimming problem starts with a field inhomogeneity in 3D space, translates this into a 1D NMR signal, after which there is no straightforward link back to the correct shim currents. Therefore, we provide the weak learners with a batch of four NMR spectra \(\mathbf{x}\) as input and predict a three-valued vector of continuous values \(\mathbf{y}\), i.e., the shim values.

Each architecture begins with \(3 - 5\) blocks of one-dimensional convolutional layers, with varying kernel sizes, followed by two fully connected layers with \(32\) nodes. Heterogeneous architectures are used to attain higher variance in predictions that are useful in the ensemble model. Additionally, we include dropout layers after convolution and fully-connected layers to prevent overfitting [32]. The last layer uses linear activation for regression of the targets. The architecture is illustrated in Fig. 4a.

Level-1. The meta-model combines features of its \(m\) heterogeneous weak learners, and represents a mixture of stacking and boosting (see subsection 2.2). We investigate the following forms:

- Simple average over all weak learner predictions.
- Non-linear combination, with a fully-connected layer, of the weak learner's regression layer (\(\in \mathbb{R}^{m\times n}\)).
- A two-layer multi-layer perceptron (MLP) based on the second-to-last fully-connected layer of the level-0 models. The MLP has \(m \times 32\) nodes in its first, and \(32\) nodes in its second layer.

Also refer to Fig. 4b for a visualization of the ensemble with an MLP-based meta-model.

4.3. Spectral quality and performance metrics

To judge the quality of spectra, we introduce a criterion \(c\) that can be used for global or local quality judgements of single peaks. For a spectrum \(g\) of interest and a reference spectrum \(r\), it is defined as:

\[
c(g, r) = \frac{1}{2} \left( \lambda_1 \cdot \frac{\text{FWHM}(r)}{\text{FWHM}(g)} + \lambda_2 \cdot \frac{\max(g)}{\max(r)} \right),
\]

where \(\max(\cdot)\) is the maximum peak height and FWHM(\(\cdot\)) is the full width at half maximum of that peak. \(\lambda_1\) can be used to increase or decrease the impact of each term. The quality parameter \(c\) indicates whether spectrum \(g\) is worse (\(0 < c < 1\)) or better (\(c > 1\)) than the reference \(r\).

Other spectral properties, such as peak symmetry, can easily extend the criterion. But since linear shims cannot lead to symmetrical line shapes, we have refrained from more precise quality measures such as the envelope of a spectral peak [52]. An extension to multiple peaks could be achieved via virtual peaks [53].

Furthermore, the following metrics are introduced to analyze the performance of our method in laboratory experiments for the \(n = 3\) first-order shims:

- Success rate SR \(\in [0, 1]\). We defined the SR w.r.t. the criterion \(c\). If the predicted shim setting yielded \(c\) higher than all \(c\) in the input batch, then the method was deemed successful. For a single experiment, it was defined as:

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Acquisition parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nr. spectra</td>
<td>9261</td>
</tr>
<tr>
<td>Shim range R</td>
<td>±10000</td>
</tr>
<tr>
<td>Step size s</td>
<td>1000</td>
</tr>
<tr>
<td>Shims</td>
<td>X, Y, Z</td>
</tr>
<tr>
<td>Nucleus</td>
<td>1H</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>20 kHz</td>
</tr>
<tr>
<td>Points</td>
<td>32768</td>
</tr>
<tr>
<td>Repetition time</td>
<td>2000 ms</td>
</tr>
<tr>
<td>Filtering</td>
<td>-</td>
</tr>
<tr>
<td>Phase correction</td>
<td>(\phi_0)</td>
</tr>
</tbody>
</table>

Table 1 Characteristics and acquisition parameters of the first-order shimming dataset (LinearShimDB).
Fig. 4. Visualization of neural network architectures. Each weak learner of the ensemble consists of \( \ell \) layers followed by two fully-connected layers. The meta-model shown here is a two-layer MLP trained based on weak learners on the transfer set \( T \).

\[
SR = \begin{cases} 
1, & \text{if } c_{\text{sh}} > \left| c_{\text{init}}, c_X, c_Y, c_Z \right| \\
0, & \text{otherwise},
\end{cases}
\]

where \( \left| c_{\text{init}}, c_X, c_Y, c_Z \right| \) are the quality values for the input batch and \( c_{\text{sh}} \) is the criterion after shimming with DRE.

- Correct direction ratio \( \text{DiR} \in [0, 1] \). Indicator whether the method pointed towards the global minimum and equals to 1 if the predicted signs matched the distortion’s signs:

\[
\text{DiR} = \frac{1}{n} \sum_{i=1}^{n} (\text{sgn}(y_i) =\text{sgn}(y_i^0)),
\]

where \( \text{sgn}(\cdot) \) is the sign function, \( y_i \) is the model’s prediction and \( y_i^0 \) is the true distortion.

- Mean improvement of criterion \( c \) (see Eq. 2) with \( i_1 = i_2 \). Given in percentage for \( c(g, r) \), where \( g \) is the spectrum with predicted correction and \( r \) is the initial spectrum.

- Averaged MAE between predictions and random distortion.

- Generalization to other substances.

5. Experiments

In this section, we describe prerequisites in pre-processing, hardware, and DL training, followed by our evaluation protocol, and the results for offline and in situ (online) experiments.

5.1. Setup or implementation details

Dataset creation and pre-processing. Each member of a batch of input spectra \( \mathbf{x} \) consists of one spectrum corresponding to a unique target value \( \mathbf{y} \), and three spectra with offsets of \( s \) in the \( X \), \( Y \), and \( Z \) shims, respectively. The database \( \mathcal{D} \) with input-target pairs \( (\mathbf{x}, \mathbf{y}) \) is constructed by mining the dataset LinearShimDB from Section 3. Each raw FID is fast-Fourier-transformed and phase-corrected to yield a 1D spectrum, using \texttt{nmrglue} [54] and the same phase correction values given by the system’s auto-phase method. Note that the unique target values \( \mathbf{y} \) for each input \( \mathbf{x} \) is not represented by its absolute shim values, but is defined by its relative distortion w.r.t. the reference spectrum of \( \mathcal{D} \). This prevents the model from learning absolute shim currents that would depend on the hardware that the database was acquired on, and forces the model to learn the relative shim offsets that will improve a given spectral shape.

In order to achieve faster convergence and generalization, our data is further pre-processed. Each spectrum is normalized by a constant normalization factor of 1e5 (the maximum intensity for perfect shims) such that the spectral values lie within the range [0, 1]. To meet resource constraints, all spectra are downsampled from 32768 to 2048 data points. The regression targets (stored as int16 integers) are divided by \( 2^{15} \) to avoid exploding gradients, and then multiplied by 100 to avoid vanishing gradients during DL model training. We exclude dataset samples where offsets are incomplete. Thus, the final subsets for training, validation, and test are of size 6400/800/801.

Due to time constraints during the acquisition of \( \mathcal{D} \) in a grid-like manner (Section 3), we expect that the spectra of each instance \( (\mathbf{x}, \mathbf{y}) \), inhibit some reality gap and temporal equalization. Therefore, we introduce an additional transfer database \( T \) with \( |T| = 100 \) to differentiate from the systematic nature of data collection. \( T \) is obtained under the same conditions as \( \mathcal{D} \), but each spectrum of \( \mathbf{x} \) is jointly acquired and is used to either fine-tune the weak learner or the meta-model.

Hardware interface. Communication with the Magrite Spinsolve spectrometer was enabled through a custom interface between Python and the python-like programming language Prosap, upon which the Spinsolve-Expert software is built. With this interface, it is possible to benefit from open-source python libraries for NMR data processing (\texttt{nmrglue} [54]) and deep learning frameworks (\texttt{PyTorch} [55], and \texttt{ray tune} [56]). The standard shimming routines used by the spectrometer’s software are based on parabolic interpolation and the downhill simplex method, and the latter is adopted for comparison.

DL training details. Level-0 base models underwent a limited neural architecture search (NAS), a technique for automating the neural network architecture design [57], using \texttt{ray tune} [56], and random search over a variable number of layers \( \ell \), kernel sizes, and other design choices. All weak learners were trained with a learning rate of 0.001, batch size of 32, and the Adam optimizer [58] for a maximum of 150 epochs utilizing early stopping. We selected the top-50 architectures among 300 runs w.r.t. validation error. The level-1 meta-models were trained with hyperparameter optimization (HPO) using random search and early stopping. We manually selected the best fully-connected and MLP-based networks w.r.t. their validation loss over 500 run. For a detailed description, see subsection S1.2.

Hardware requirements. The DL training was performed with an AMD Ryzen 5900X equipped with 64 GB RAM, and a graphics processing unit NVIDIA GeForce RTX 3090. The LinearShimDB roughly requires 3.4GB of disc space.

5.2. In situ evaluation protocol

We demonstrate in situ functionality by testing the method on a set of 100 random distortions \( y_x, y_y, y_z \in [-10000, 10000] \) of \( X \), \( Y \), \( Z \) shims, drawn from a uniform distribution. We deliver success rate \( SR \), direction rate \( \text{DiR} \), mean improvement of our quality parameter \( c \), and MAE over five different model types, including single weak
learners and diverse ensembles. Furthermore, we investigate generalizability towards multiple samples, i.e., pure water with CuSO₄ as in Section 3, ethanol (CAS No. 67–66-3) dissolved in water with a molar fraction of χ = 0.1-0.5, and isopropanol (CAS No. 67–63-0) in water with χ = 0.5.

5.3. Level-0 training results

Offline experiments, i.e., training on static data, show that it is possible to predict three distinct variables from an input of four 1-D signals with no apparent correlation between the input and output dimensions. The best weak learner achieved an MAE of 596 ± 769 (mean ± standard deviation) for a step size of s = 1000 on the test set. As the possible precision was limited by the sampling resolution of the underlying training data, results near 1/2 of the step size s indicated good performance. Detailed results are given in Table S1. The entire network, including the meta-model, was tested in situ only.

5.4. In situ results

The most crucial aspect of the method is not merely its training performance, but also its applicability. Thus, the performance of our method is evaluated in laboratory experiments, and a selection of graphical results is given in Table 3 for our most promising methods (single model and MLP-based) and different substances (H₂O, ethanol and isopropanol).

Interpretation of different model types. The results seem to indicate that our method works in practise. Even weak learners achieved an SR of 93% and a large improvement (mean of +435%) on the spectral quality for water. However, the variance in criterion improvement and error remained high. Here, the ensemble method with an MLP-based meta-model achieved more robust but conservative predictions. Overall, a single model and the MLP-based ensemble yield comparable results for all measured metrics.

Absent improvement by averaging the top 50 untuned models confirmed the need to train a meta-model. Furthermore, the advantage of a two-layer MLP with non-linear connections of the second-to-last features is shown over a simple non-linear combination of the weak learner’s last layer. We also fine-tuned a single model to the transfer dataset T, yielding worse in situ results. Overall, a transfer set seems to be unnecessary.

The robustness problem of a single model compared to the MLP-based ensemble is visualized in Fig. 5 based on the experiments that yield Table 2 for water. The single model can often achieve narrower linewidths but shows higher variance.

5.5. Comparison

We compare our automated shimming method to Magritek’s built-in implementation of the downhill simplex method, which in turn is based on the algorithm in [50].

We expect faster convergence of the simplex method when using the improvements as proposed by [12]. Nevertheless, Yao et al. state that their method behaves similarly to the simplex method when fewer shim coils are used, so we restricted our considerations to the Magritek implementation of the regular downhill simplex method. This implementation requires at least n + 1 measurements for n shims to initialize its simplex structure, and one to four (average of two) function evaluations per iteration [11]. Our method consistently needs four spectra for one iteration, and one spectrum to check the results. We used a single model and DRE with MLP as the meta-model for comparison.

Although the downhill simplex is known to have little influence on the initial simplex’ size and shape [11], and tends to produce rapid drops of initial values [59], we were able to accelerate the shimming process. We compared results obtained from the standard Nelder-Mead method with i iterations and step size s = 1000, versus the results for simplex with i – 4 iterations while

Table 2

<table>
<thead>
<tr>
<th>Substance</th>
<th>Single model</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Untuned</td>
<td>Tuned to T</td>
</tr>
<tr>
<td>H₂O</td>
<td>Success Rate SR</td>
<td>0.93</td>
</tr>
<tr>
<td></td>
<td>Direction Ratio DiR</td>
<td>0.90</td>
</tr>
<tr>
<td></td>
<td>Mean improvement of c</td>
<td>+435%±559</td>
</tr>
<tr>
<td></td>
<td>Averaged MAE</td>
<td>1878±1277</td>
</tr>
<tr>
<td>Ethanol (0.5)</td>
<td>SR</td>
<td>0.92</td>
</tr>
<tr>
<td></td>
<td>DiR</td>
<td>0.92</td>
</tr>
<tr>
<td></td>
<td>Mean improvement of c</td>
<td>157%±130</td>
</tr>
<tr>
<td></td>
<td>Averaged MAE</td>
<td>2431±1287</td>
</tr>
<tr>
<td>Ethanol (0.1)</td>
<td>SR</td>
<td>0.91</td>
</tr>
<tr>
<td></td>
<td>DiR</td>
<td>0.87</td>
</tr>
<tr>
<td></td>
<td>Mean improvement of c</td>
<td>+265%±214</td>
</tr>
<tr>
<td></td>
<td>Averaged MAE</td>
<td>2170±1292</td>
</tr>
<tr>
<td>Isopropanol (0.5)</td>
<td>SR</td>
<td>0.91</td>
</tr>
<tr>
<td></td>
<td>DiR</td>
<td>0.89</td>
</tr>
<tr>
<td></td>
<td>Mean improvement of c</td>
<td>+299%±320</td>
</tr>
<tr>
<td></td>
<td>Averaged MAE</td>
<td>1967±1177</td>
</tr>
</tbody>
</table>
We reduce iterations by four because one iteration of DRE needs four spectra for its prediction. Furthermore, we compared the number of function evaluations necessary for the simplex to reach a criterion equivalent to the DRE prediction, as shown in Table 5. The procedure is as follows: First, our method "deep regression with ensembles" (DRE) predicts shim settings for a random distortion drawn from a uniform distribution. Then, the downhill simplex method is started from the same distortion with a step size of $s = 1000$ (as used for the database in Section 3). The algorithm is stopped when it reaches a linewidth equivalent to the one achieved with DRE prediction, and the number of acquisitions (function evaluations) are reported. If the simplex is not able to find an equivalent linewidth within 50 iterations, it is stopped.

The results in Tables 4 and 5 demonstrate that, using our method as stand-alone or in combination with regular shimming methods will provide an advantage in either the number of necessary acquisitions, or the achieved spectral quality. Note that deep regression (DR) apparently yields similar performance to deep regression with ensembles (DRE) in quality improvement, but initialized with our method (see Table 4).
ensembles demonstrate advantage in the number of necessary NMR acquisitions as compared to the simplex method.

6. Discussion

Discovering new methods for fast and efficient NMR shim coil calibration remains a challenge, but we see great potential in a fusion of traditional and modern algorithmic methods to achieve both faster and more precise shimming.

Assumptions of our approach. Currently, our approach mimics a scenario where only the first-order shims are available to shim a probe from scratch. Thus, the best achievable FWHM is of the order of tens of Hz, but our method shows that very broad initial lineshapes can be improved, very nearly reaching the optimum. Furthermore, using linear shims is the first step for DL to advance shimming for more complex and non-standard cases (e.g. high-throughput parallel spectroscopy where some samples may be located off-center with respect to the shim system, shimming of micro coils). Although first-order shims generally have a prominent influence on reducing inhomogeneity, dealing with higher-order shims should be considered in future work. We are also convinced that localizable information, e.g. about axial magnetization [17], with special hardware additions could help the DL method and allow to reduce the number of required input spectra. Our method contrasts with traditional shimming approaches, in that it is not iterative, i.e., the result emerges after a single step. Therefore, we cannot guarantee that iterating DR or DRE will converge to better results under other circumstances.

Deep learning considerations. Also, it is unclear how or whether supervised learning can be scaled for shimming of higher-order shims, especially w.r.t. exponentially increasing requirements in the data acquisition task. One approach to counteract this limitation is to employ a sufficiently realistic digital twin (simulation) of the shimming problem to generate synthetic data.

In any case, this study has not yet unlocked the full potential of deep learning, as applied to the shimming problem in NMR. Especially, recent advances in explainable artificial intelligence (XAI) [60] can help to understand decisions made by neural networks, or new neural architectures such as transformers [61,62] are creating a big fuss and yield excellent performance in various tasks. However, to stay grounded in the complexity of our method, we expect higher stability of a single model by using benchmark architectures such as ResNet [63], and increased efficiency by realizing other deep learning techniques [64].

Despite the limitation described above, we could prove that the concept of using DL for NMR shimming is a promising approach to accelerate the entire shimming process.

In general, we also wish to draw attention to the lack of published code for related shimming methods, and the difficulty of comparing published results obtained on diverse spectrometers and hardware setups.

7. Conclusion

In this paper, we have conclusively shown that deep learning can be used for fast, first-order shimming in a low-field NMR setup, i.e., DL can tackle the ambiguity and lineshape problems inherited by shimming. With our dataset for shimming, we furthermore established a basis for continued research in this area. The applicability of DL was demonstrated both offline and in situ: First, by training neural networks to simultaneously predict three linear-field shim currents necessary to partially cancel an arbitrary distortion. And second, the deployment on a spectrometer of a meta-model based on an ensemble of weak learners revealed that our non-iterative method generally points towards the solution and shows a high success rate in improving spectral quality. Despite the models being trained entirely on water, we experienced generalizability to more complex samples with more than one spectral peak.

Author contributions

J.K. acquired the funding, and together with M.J., conceived the idea, and performed supervision. M.B. generated the database and designed the DL algorithm. M.B. conceived and performed the numerical experiments. M.B. prepared the data tables and plots, and wrote the initial manuscript. M.B., A.K., M.J. and J.K. iterated the manuscript, and proof read the manuscript. All authors participated in the discussions.

Financial disclosure

J.G.K. declares financial interest in the startup company Voxel analytic GmbH, which develops and sells NMR equipment. The other authors declare no interest.

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.

Acknowledgments

We thank Craig Eccles of Magritek for his great support. Plots in Fig. 3, Table 3 and Fig. 5 where made with the Python library Sci- encePlots [65], and we sincerely thank John Garrett for making the code available online. We acknowledge support by the KIT-Publication Fund of the Karlsruhe Institute of Technology.

Appendix A. Supplementary material

The following supporting information is available as part of the online article: Supplementary at the end of this document, data-

References


