Unrolled primal-dual deep network for sparse signal restoration

Mouna Gharbi, Silvia Villa, Emilie Chouzenoux, Jean-Christophe Pesquet

Unrolled primal-dual deep network for sparse signal restoration

Mouna GHARBI[†], Silvia VILLA[*], Emilie CHOUZENOUX[†], Jean-Christophe PESQUET[†]

***Abstract*— This paper addresses the problem of sparse data recovery. This is an important task in many signal and image processing areas, and is often solved via ℓ_1 regularization. Nowadays, many efficient algorithms are available to solve the corresponding optimization problem. While their convergence is based on a well established mathematical theory, their practical perfomance is heavily determined by the selection of appropriate hyperparameters (e.g. the stepsize). This choice is often difficult and time consuming. In this paper we propose a deep network architecture based on the unrolling of a primal-dual algorithm. This allows us to learn the hyperparameters involved in the algorithm automatically in a data driven way. The proposed network has an interpretable structure where each layer mimics one iteration of the primal-dual algorithm. The method is computationally efficient thanks to deep learning framework accelerations. Through an example arising from spectroscopy signal restoration, we show that unrolling improves the restoration performance with respect to state-of-the-art, including the iterative implementation of primal-dual algorithm.**

I. INTRODUCTION

Sparse signal restoration is a common inverse problem arising in machine learning, spectroscopy, imaging, etc. The goal is to retrieve original useful data given observations corrupted in experiments, under the assumption that the sought solution is a sparse vector characterised by a small number of nonzero entries. Acquired signals are related to groundtruth data through the model

$$z = H\bar{x} + \varepsilon, \tag{1}$$

where $z \in \mathbb{R}^m$ is a degraded acquisition of the original signal $\bar{x} \in \mathbb{R}^n$, $H \in \mathbb{R}^{m \times n}$ is a linear operator simulating the acquisition model, for instance a convolution, and $\varepsilon \in \mathbb{R}^m$ is the realization of an additive zero-mean i.i.d. Gaussian noise. The problem of estimating $\bar{x}$ from the degraded observation z is called signal recovery [1] and several approaches in literature have been proposed to solve it. A known approach consists in using iterative algorithms to minimize a designed objective. For instance, in variational regularization or explicit regularization [2], a cost function composed of a data fidelity term and a prior term balanced by a regularization hyperparameter is minimized. Regularization is incorporated into the problem to mend the ill-posedness of the problem [3]. The penalization is designed to enforce sparsity in the recovered estimate. For instance, ℓ_1 norm, ℓ_0 quasi-norm or their smooth approximations can be used. Another type of regularization consists in iterative or implicit regularization [4] in which the implicit bias of first-order algorithms is exploited to favor desirable properties of the solution. In this case, the number of iterations plays the role of the regularization parameter.

Solving inverse problems of the form (1) has also been addressed by supervised learning methods. In fact, common deep network architectures (such as CNNs) have been used for deblurring [5], denoising [6] and signal reconstruction [7]. Such models are trained on large annotated databases, to learn a mapping between the input (degraded data) and output (original data). This mapping is then used on test data. Deep learning approaches have proven their efficiency when provided with enough data. They are also fast and easy to manipulate since they do not require designing models built on any prior knowledge. However, their robustness, stability and interpretability are still active research directions, which explains why they are often described as black box models.

Lately, a new approach called unrolling/unfolding, combining model-based and data-driven approaches has emerged. The basic idea consists in creating deep network architectures inspired from iterative algorithms, such that each layer mirrors one iteration of the corresponding algorithm. The benefits of this method are threefold, namely (i) its interpretability which contrasts with black-box mainstream models, (ii) its speed thanks to deep learning frameworks such as Pytorch and Tensorflow, (iii) its efficiency since unrolled architectures allow automatic tuning of the algorithm native hyperparameters using supervised learning techniques and built in utilities such as backpropagation and differential programming. Many scientific works have explored unrolling of various algorithms, such as gradient descent [8], ISTA [9], and proximal interior methods [10]. In [11], learnt denoisers were involved in a half-quadratic splitting scheme for inverse problem resolution. Finally, [12] investigated the unrolling of a variational Bayesien algorithm and [13] a block dual forward-backward algorithm.

Several recent works have unrolled primal-dual proximal splitting algorithm [14] for various inverse problem tasks. Namely, for MRI reconstruction [15] the proposed method PD-net learns operators associated to the proximity of regularization and data fitting terms, while implicitly including stepsizes and extrapolation. Similarly, for a tomographic reconstruction in [16], CNNs are used to learn operators

This work has been supported by the ITN-ETN project TraDE-OPT funded by the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 861137. M.G. and E.C. acknowledge funding support from the European Research Council Starting Grant MAJORIS ERC-2019-STG850925. S.V. acknoledges the support of the project AFOSR (European Office of Aerospace Research and Development) FA8655-22-1-7034, the European Research Council Consolidator Grant SLING 819789 and the EU H2020-MSCA-RISE project NoMADS-777826.

[†] CVN, CentraleSupélec, Inria Saclay, University Paris Saclay, France.

[*] MaLGa, DIMA, Università degli Studi di Genova, Via Dodecaneso 35, 16146 Genova, Italy.

that implicitly encode proximity operators, stepsizes and extrapolation hyperparameters while allowing it to vary along layers and using a memory-based approach to reuse old primal iterates. In [17], the learnt parameters are the algorithm step-sizes and the analysis linear operator involved in the penalization including regularization. In [18], the linear operator involved in the penalization and the prox associated to the regularization are learnt though CNNs while the primal, dual stepsizes, scale and extrapolation hyperparamaters seem to be fixed manually. In this work, our contribution is the unrolling of a primal-dual algorithm using a constrained formulation inspired from the implicit regularization framework to reconstruct sparse signals. First, this yields a fast method thanks to the GPU tools deployed by deep learning frameworks, Second, this allows to learn stepsize and noise-related hyper-parameters of the original algorithm automatically to reach optimal signal restoration. The proposed method is compared to various other methods used for sparse signal restoration.

The paper is organized as follows: in section II, we introduce the mathematical notation and definitions useful for the rest of the work. Section III presents our main contribution starting from modeling the problem and devising a primal-dual algorithm to solve it, to unrolling the latter. We also detail our strategy to learn the desired hyperparameters. Section IV summarizes our experimental settings, comparisons and analysis. Lastly, section V draws conclusions of the work.

II. Mathematical notation

We adopt convex analysis notations from [19]. Let $\mathscr{H}$ be a Hilbert space, the subdifferential of a function $f:\mathscr{H}\to\mathbb{R}$ at point x is the set valued operator $\partial f:\mathscr{H}\to 2^{\mathscr{H}}$ such that $\partial f(x):\{u\in\mathscr{H}|(\forall y\in\mathscr{H})\langle y-x|u\rangle+f(x)\leq f(y)\}$. The identity operator on $\mathscr{H}$ is denoted by Id. The set of proper, convex and lower semi continuous (l.s.c) functions is denoted by $\Gamma_0(\mathscr{H})$ and the proximity operator is defined for every $x\in\mathscr{H}$ by $\text{prox}_f(x)=\text{argmin}_{u\in\mathscr{H}} f(u)+\frac{1}{2}\|x-u\|_2^2$. Finally, the convex conjugate of a function is denoted, for every $x\in\mathscr{H}$, by $f^*(x)=\sup_{u\in\mathscr{H}}\langle x|u\rangle-f(u)$. For a given convex closed non empty set C, ι_C is the indicator function defined by

$$(\forall x\in\mathscr{H})\quad \iota_C(x)=\begin{cases}0, & \text{if}\quad x\in C\\ +\infty & \text{otherwise.}\end{cases} \tag{2}$$

The support function $\sigma_C=\iota_C^*$ is defined by

$$(\forall x\in\mathscr{H})\qquad \sigma_C(x)=\sup_{x'\in C}\langle x'|x\rangle. \tag{3}$$

We finally denote by $\mathscr{B}(z,\rho)$ the ℓ_2 ball of center z and radius ρ,

$$\mathscr{B}(z,\rho)=\{z'\in\mathscr{H}\mid\|z'-z\|_2\leq\rho\}. \tag{4}$$

III. Proposed method

A. Problem statement

Let us define $\hat{x}\in\mathbb{R}^n$ an estimate of $\bar{x}$ obtained by solving the constrained minimization problem

$$\hat{x}\in\underset{x\in\mathbb{R}^n}{\text{argmin}}\,\|x\|_1\quad s.t.\quad\|Hx-z\|_2\leq\rho. \tag{5}$$

Hereabove, $\|\cdot\|_1$ is ℓ_1 norm which aims to enforce a sparsity prior on the estimated solution and $\rho>0$ is a hyperparameter related to the noise level. In the case of an i.i.d. Gaussian noise with standard deviation σ, a common setting arising from the discrepancy principle is $\rho=\sqrt{n}\sigma$. The minimization problem (5) can be solved using a proximal splitting primal-dual algorithm [20], [14], [21] which we describe in the next section. The optimal solution is retrieved combining the explicit regularization induced by ρ with the implicit regularization given by the early stopping of the iterations [22].

B. Primal-dual algorithm

Problem (5) can be recast into an unconstrained form which is the primal problem

$$\min_{x\in\mathbb{R}^n}\quad\Big(\iota_{\mathscr{B}(z,\rho)}(Hx)+\|x\|_1\Big), \tag{6}$$

The associated dual problem [20], [23] is

$$\max_{y\in\mathbb{R}^m}\quad-\Big(\sigma_{\mathscr{B}(z,\rho)}(y)+\|-H^\top y\|_1\Big). \tag{7}$$

The general saddle point problem is

$$\min_{x\in\mathbb{R}^n}\max_{y\in\mathbb{R}^m}\quad\langle Hx,y\rangle+\|x\|_1-\sigma_{\mathscr{B}(z,\rho)}(y). \tag{8}$$

To solve the saddle point problem, the primal-dual approach initially proposed in [20] can be used, leading to Algorithm 1. Convergence of the sequence $(x_k)_{k\in\mathbb{N}}$ to a solution of (5) is established in [21], for γ,τ positive stepsizes verifying $\gamma\tau\leq\frac{1}{\|H\|_2^2}$.

Algorithm 1 Primal-dual algorithm

Init: Choose $\tau,\gamma>0$, $(x_0,y_0)\in\mathbb{R}^n\times\mathbb{R}^m$ and $y_0=y_{-1}$.
for $k=0,1,\ldots$ **do**
  $\tilde{y}_k=2y_k-y_{k-1}$
  $x_{k+1}=\text{prox}_{\tau\|\cdot\|_1}(x_k-\tau H^\top\tilde{y}_k)$
  $y_{k+1}=\text{prox}_{\gamma\sigma_{\mathscr{B}(z,\rho)}(\cdot)}(y_k+\gamma Hx_{k+1})$
end for

We now give explicitly the expressions of the proximity operators involved in Algorithm 1. First,

$$(\forall x=(x_i)_{1\leq i\leq n}\in\mathbb{R}^n)\qquad\text{prox}_{\tau\|\cdot\|_1}(x)=(\text{sign}(x_i)\max(|x_i|-\tau,0))_{1\leq i\leq n}. \tag{9}$$

Second, using Moreau's decomposition in [19, Theorem 14.3 (ii)], for every $\gamma>0$,

$$(\forall y\in\mathbb{R}^m)\quad\text{prox}_{\gamma\sigma_{\mathscr{B}(z,\rho)}}(y)=y-\gamma P_{\mathscr{B}(z,\rho)}\left(\gamma^{-1}y\right). \tag{10}$$

Hereabove, $P_{\mathscr{B}(z,\rho)}$ is the projection onto the ball $\mathscr{B}(z,\rho)$ of center z and radius ρ,

$$(\forall y\in\mathbb{R}^m)\,P_{\mathscr{B}(z,\rho)}(y)=\begin{cases}y & \text{if}\quad y\in\mathscr{B}(z,\rho)\\ z+\frac{y-z}{\|y-z\|_2}\rho & \text{if}\quad y\notin\mathscr{B}(z,\rho)\end{cases}. \tag{11}$$

Finally, this leads, for every $y \in \mathbb{R}^m$, and for every $\gamma > 0$, to

$$\text{prox}_{\gamma\sigma_{\mathscr{B}(z,\rho)}}(y) = \begin{cases} 0, & \text{if} \quad \gamma^{-1}y \in \mathscr{B}(z,\rho) \\ y - \gamma(z + \frac{y-\gamma z}{\|y-\gamma z\|_2}\rho) & \text{if} \quad \gamma^{-1}y \notin \mathscr{B}(z,\rho) \end{cases}. \tag{12}$$

Setting the stepsize hyperparameters and the noise control hyperparameter is usually a time consuming task when performed manually, and is often the computational bottleneck of the regularization approach. To this end we propose an unrolling strategy allowing their automatic tuning and an enhanced restoration quality.

C. Unrolled primal-dual algorithm

To create a deep neural network architecture, unrolling starts by converting, for a fixed number of layers K, the concerned algorithm iterations into layers. In our case, the primal and dual iterations in Algorithm 1 reveal building blocks of feedforward network structures. Therefore, we propose the following multi-branch architecture: for $k \in \{0,...,K-1\}$, the primal $\mathscr{L}_{p_k}$ and dual $\mathscr{L}^z_{d_k}$ branches process respectively the sequences $(x_k)_{0\le k\le K}$ and $(y_k)_{0\le k\le K}$ as follows:

$$\Big(x_{k+1}, y_{k+1}\Big) = \Big(\mathscr{L}_{p_k}(x_k, y_k, y_{k-1}), \mathscr{L}^z_{d_k}(\mathscr{L}_{p_k}(x_k, y_k, y_{k-1}), y_k)\Big). \tag{13}$$

For the sake of short notation we denote by $\mathscr{L}^z_k$, for $k \in \{0,...,K-1\}$, a global layer of our proposed architecture. It depends on the noisy observation given the dependence of the dual branch on z. It takes as input the couple (x_y, y_k) and returns as output the couple (x_{k+1}, y_{k+1}) except the last layer which only returns x_K. In that case, (13) becomes

$$\Big(x_{k+1}, y_{k+1}\Big) = \mathscr{L}^z_k(x_k, y_k) \tag{14}$$

such that,

$$\begin{cases} x_{k+1} = \mathscr{A}_{p_k}(W_{p_k}x_k + b_{p_k}) \\ y_{k+1} = \mathscr{A}_{dk}(W_{dk}y_k + b_{dk}) \end{cases}. \tag{15}$$

Here, $\mathscr{A}_{p_k}$ and $\mathscr{A}_{dk}$ are primal and dual activation functions:

$$(\forall x \in \mathbb{R}^n) \quad \mathscr{A}_{p_k}(x) = \text{prox}_{\tau_k\|\cdot\|_1}(x), \tag{16}$$

$$(\forall y \in \mathbb{R}^m) \quad \mathscr{A}_{dk}(y) = \text{prox}_{\gamma_k\sigma_{\mathscr{B}(z,\rho)}(\cdot)}(y). \tag{17}$$

So as to retrieve equivalence with our Algorithm 1, we set the weight matrices W_{p_k} and W_{dk} equal to the identity of their respective ambient space and bias terms are given by

$$b_{p_k} = -\tau_k H^\top \tilde{y}_k \in \mathbb{R}^n, \tag{18}$$

$$b_{dk} = \gamma_k H x_{k+1} \in \mathbb{R}^m. \tag{19}$$

As can be seen in the aforementioned equations (16)-(18), we propose to untie sequences $(\tau_k)_{0\le k\le K-1}$, $(\gamma_k)_{0\le k\le K-1}$ and $(\rho_k)_{0\le k\le K-1}$ through unrolling, and allow them to vary along layers. Furthermore, we propose to learn these sequences automatically. To this end, we design simple network architectures which will be reattached to each layer $\mathscr{L}^z_k$. We propose to enforce positivity on learnable parameters through the ReLU [24] activation function

$$(\forall z = (z_\ell)_{1\le\ell\le m} \in \mathbb{R}^m) \quad \text{ReLU}(z) = (\max(0, z_\ell))_{1\le\ell\le m}. \tag{20}$$

Specifically, let $\theta = \{t_k, s_k, r_k\}_{0\le k\le K-1}$ be the weights to be learned then for all $k \in \{0,\dots,K-1\}$, hyperparameters will be given by

$$\tau_k = \text{ReLU}(t_k), \quad \gamma_k = \text{ReLU}(g_k), \quad \rho_k = \text{ReLU}(r_k). \tag{21}$$

Once the multi-layer architecture is defined, updating the learnable weights is a straightforward task. Let $\mathscr{S} = \{(\bar{x}_s, z_s) | s \in \{1,\dots,S\}\}$ be a training set comprised of S pairs of groundtruth and degraded data samples. Let the prediction function be

$$g^z_\theta(x_{s,0}, y_{s,0}) = \mathscr{L}^z_{K-1} \circ \dots \circ \mathscr{L}^z_k \circ \dots \circ \mathscr{L}^z_0(x_{s,0}, y_{s,0}), \tag{22}$$

$(x_{s,0}, y_{s,0})$ denoting the primal and dual inputs for the first layer of the network for sample s. Then, for a predefined loss ℓ, the optimal hyperparameter setting $\hat{\theta}$ is the solution of the optimization problem

$$\min_\theta \quad E(\theta) = \frac{1}{S}\sum_{s=1}^{S} \ell(g^{z_s}_\theta(x_{s,0}, y_{s,0}), \bar{x}_s). \tag{23}$$

The mean squared error (MSE) loss is obtained when setting $\ell(x,\bar{x}) = \frac{1}{n}\|x-\bar{x}\|^2$. Standard neural network training algorithms such as stochastic gradient or ADAM can be used to solve the above problem.

IV. Experimental results

In this section we present our experimental results through an application arising from mass spectrometry (MS) [25] signal restoration. MS is a chemical tool used for detection and quantification of molecules of interest. It produces spectra with positively valued peaks distributed according to the isotopic distribution of the studied molecule and its charge. We consider here the problem of deblurring and denoising MS spectra.

A. Datasets

We test our proposed unrolled method on two datasets of realistic MS signals, generated using the Averagine model [26]. First, for a given number of proteins $p = 10$, a 2000 dimensional signal with p randomly uniformly placed peaks between index 50 and 1950 is created. The associated positive intensities are randomly defined as the absolute value of a Gaussian realization with mean 10 and standard deviation 100. This signal is later multiplied by the MS averagine dictionary gathering isotopic distribution of a predefined list of atomic mass. This results in the groundtruth signal $\bar{x}$. Model (1) is then used to create the associated degraded signal z. The convolutional blur kernel h is a centered Gaussian shape with support size 50 and standard deviation 1. It is implemented through the Toeplitz matrix $H \in \mathbb{R}^{m\times n}$, with $n = 2000$ and $m = 2049$ mimicking the circular padded convolution. For Dataset 1, the added Gaussian noise $\varepsilon \in \mathbb{R}^m$ is i.i.d., zero mean with standard deviation $\sigma = 2$. From the

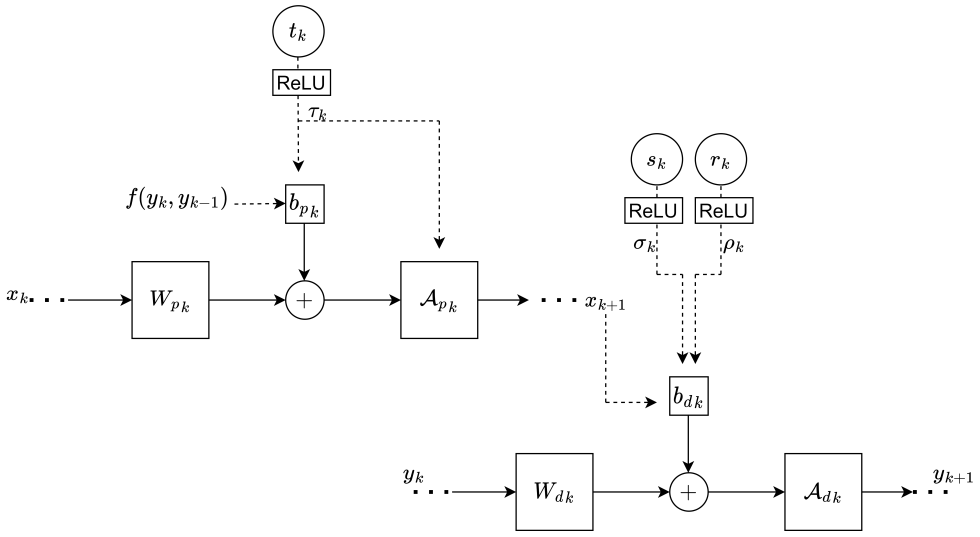

Fig. 1. Overview of one layer $\mathscr{L}_k^z$ of the proposed unrolled primal dual algorithm. $(W_{pk})_{1\leq k\leq K}$, $(W_{dk})_{1\leq k\leq K}$, $(b_{pk})_{1\leq k\leq K}$, $(b_{dk})_{1\leq k\leq K}$, $(\mathscr{A}_{pk})_{1\leq k\leq K}$ and $(\mathscr{A}_{dk})_{1\leq k\leq K}$ are respectively primal/dual weight operators, biases and activations. $(s_k)_{1\leq k\leq K}$, $(r_k)_{1\leq k\leq K}$ and $(t_k)_{1\leq k\leq K}$ are learnable weights and ReLU is an activation enforcing positivity.

same set of groundtruth signals $\bar{x}$, we furthermore create Dataset 2, where z is built assuming an i.i.d. zero mean Gaussian noise with standard deviation σ randomly chosen between 0 and 2. We create respectively a training, validation and test set of sizes 1000, 200 and 100 for both datasets.

B. Comparative methods

We compare our method to several learning-based and iterative optimization approaches. First, we run the primal-dual algorithm described in Algorithm 1. The stepsizes γ, τ are tuned following the strategy proposed in [22] such that $\tau\gamma = \frac{0.99}{\|H\|^2}$ and $\tau = \gamma$. The bound ρ is finetuned manually. Specifically, for Dataset 1, we assume the noise level to be known, and we tune ρ using a predefined grid of values including $\sigma\sqrt{n}$. For Dataset 2, as the noise level is varying, we propose to estimate it through $\hat{\sigma}(y) = \frac{\text{median}(|W_H y|)}{0.6745}$ [27] for each degraded observation y with $|W_H y|$ the vector gathering the absolute value of the diagonal coefficients of the first level Haar wavelet decomposition of y. We then set $\rho = \hat{\sigma}(y)\tilde{\rho}$ with $\tilde{\rho}$ finetuned through gridsearch. For initialization we use $x_0 = y_0 = y_{-1}$. The set of parameters providing the lowest MSE on the training set is retained, and per-used on the test set. Next, we compare our method to the iterative soft threshholding algorithm (ISTA) [28], which minimizes $x \to \frac{1}{2}\|Hx - z\|^2 + \chi\|x\|_1$ where $\chi > 0$ is a regularization hyperparameter balancing the data fidelity and regularization term. The update rule of ISTA is

$$(\forall k \in \mathbb{N}) \quad x_{k+1} = \text{prox}_{\gamma\chi\|\cdot\|_1}(x_k - \gamma H^\top(Hx_k - z)), \quad (24)$$

with γ a stepsize hyperparameter verifying $\gamma \in]0, \frac{2}{\|H\|^2}[$ and $x_0 = 0$. Hereagain, we adopt different strategies for Dataset 1 or 2. Namely, we directly finetuned χ for the former, while we set $\chi = \tilde{\chi}\hat{\sigma}(y)$ with $\tilde{\chi}$ to be tuned for the latter. The last iterative-based method we compare with, is the half quadratic (HQ) algorithm [29] used to minimize $x \to \frac{1}{2}\|Hx - z\|^2 + \lambda\sum_{i=1}^n \delta(|x_i| - \delta\log(\frac{|x_i|}{\delta} + 1))$. The penalty, known as the Fair potential, is an approximation of the ℓ_1 norm depending on a smoothing hyperparameter $\delta > 0$. We employ gridsearch to determine the described parameters (λ, δ). For learning-based methods, we provide comparisons with the unrolled versions of the described algorithms. For unrolled ISTA we learn sequences $(\chi_k)_{1\leq k\leq K}$ and $(\gamma_k)_{\{1\leq k\leq K\}}$ through the feed-forward structure

$$x_{k+1} = R_1(x_k - R_0(V_0 x_k + b_0)). \quad (25)$$

The operator $V_0 = \gamma_k H^\top H$ is a linear weight, $b_0 = -\gamma_k H^\top z$ is a bias term and $(\forall x \in \mathbb{R}^n)$ $R_0(x) = x$, $R_1(x) = \text{prox}_{\gamma_k\chi_k\|\cdot\|_1}(x)$ are the underying activation functions. As for unrolled HQ, we learn $(\lambda_k)_{1\leq k\leq K}$ according to the paradigm described in [29].

C. Training settings

We detail here the experimental settings used during the unrolling of the aforementioned described methods. The MSE loss is used for training all unrolled methods as well as to finetune the parameters of the optimization-based methods. In all experiments, the training and validation batch sizes are set to 5. Learnable weights are tuned through backpropagation using Adam [30], where the learning rate (lr) is set according to experiment to maintain a stable training. As entry for the multiple architectures we use the null vector. For unrolled HQ and ISTA, the input is $x_0 = 0$ and for unrolled primal dual $x_0 = y_0 = y_{-1} = 0$. To set the number of layers, we train the unrolled architectures for several choices of K. Then we choose the number of layers that returns a minimal loss on the validation set while maintaining a reasonable training execution time. We show the validation loss obtained as a function of K in Fig. 2 for our unrolled primal-dual method using Dataset 1. We summarize in Table I the learning rate and the number of layers used for each experiment.

Dataset1		Dataset2	
Unrolled HQ			
Adam, lr=10^{-2}	8	Adam, lr=10^{-2}	8
Unrolled ISTA			
Adam, lr= 10^{-4}	14	Adam, lr= 10^{-3}	14
Unrolled Primal Dual			
Adam, lr=10^{-4}	22	Adam, lr=10^{-4}	16

TABLE I

SUMMARY OF TRAINING SETTINGS. FOR EACH DATASET, 1^{st} COLUMN IS THE LEARNING RATE AND 2^{nd} COLUMN IS THE NUMBER OF LAYERS.

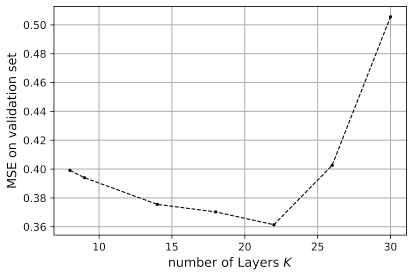

Fig. 2. Evolution of average validation loss for different number of layers, using unrolled primal-dual algorithm on Dataset 1.

D. Numerical results

Let us now present our numerical results. The mean and standard deviation of the MSE on the test set for each method is reported in Table II. A first observation is that all unrolled methods outperform their iterative counterparts in terms of restoration quality (averaged MSE) and performance dispersion (standard deviation of MSE). This result is expected since unrolled architectures learn hyperparameters, including the number of layers (i.e., iterations) from the data at hand, offering more flexibility to the iterates path. Our proposed unrolled method outperforms all the iterative methods (including Alg. 1), with a much lower computational cost (5000 iterations for primal-dual to converge vs 22 layers using unrolled primal-dual algorithm). Both unrolled primal-dual and unrolled ISTA reach a similar performance which is expected by their equivalent formulations. The advantages of our unrolled architecture are as follows. First, flexibility, which allows learnt parameters to vary from one layer to another. We display on Fig. 3 the learnt parameters along layers, with respect to their initializations for our proposed unrolled primal-dual approach. Second, the proposed architecture is capable of dealing with a dataset with a variable noise level despite the simplicity of the architecture, as it is designed to capture the noise level. Third, backpropagation allows to learn automatically the hyperparameters, which is confirmed by the learning curves on Fig. 4 of our unrolled primal-dual architecture. During supervised learning, loss over training and validation sets decreases through epochs until reaching an optimal setting. This is done in an automated manner and lifts the tedious task of tuning these hyperparameters manually. Finally, to illustrate the good quality of our results, we show a reconstruction example in Fig. 5.

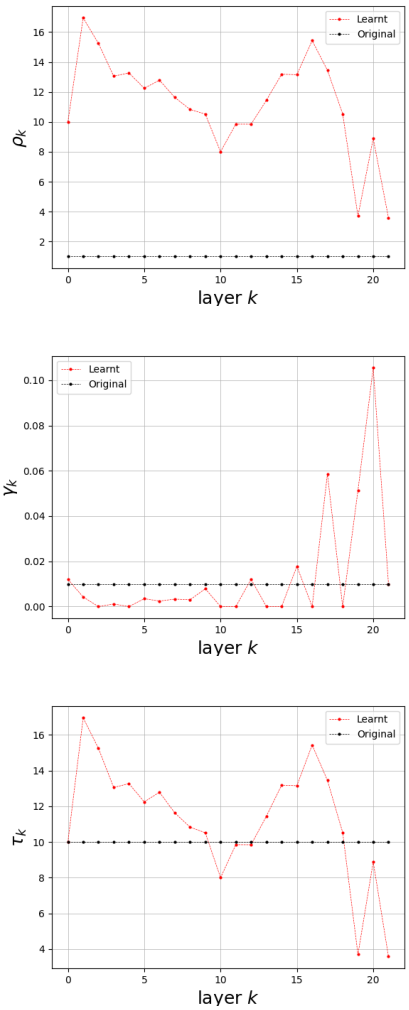

Fig. 3. Learnt hyperparameter sequences $(\rho_k)_{1\leq k\leq K}$, $(\gamma_k)_{1\leq k\leq K}$ and $(\tau_k)_{1\leq k\leq K}$ respectively for unrolled primal-dual algorithm across layers.

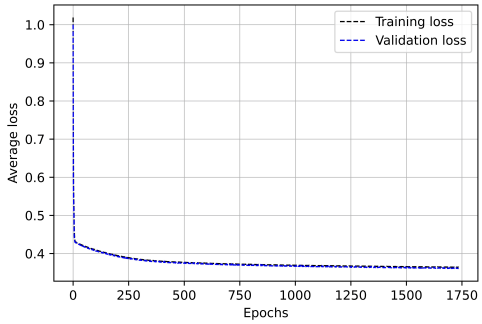

Fig. 4. Evolution of averaged training and validation MSE loss during learning using Dataset 1.

	Primal-dual	ISTA	HQ	Unrolled Primal-dual	Unrolled ISTA	Unrolled HQ
Dataset 1	3.1428 (1.1538)	3.0691 (1.1597)	0.9247 (0.2772)	0.3452 (0.1042)	0.3192 (0.0816)	0.4159 (0.0896)
Dataset 2	1.7676 (1.0201)	1.8452 (1.1693)	0.6212 (0.1948)	0.1865 (0.1125)	0.1561 (0.0923)	0.2039 (0.1197)

TABLE II

SUMMARY TABLE OF COMPARATIVE METHODS PERFORMANCE IN TERMS OF AVERAGE MSE (STANDARD DEVIATION) FOR DATASET 1 AND DATASET 2 RESPECTIVELY ON FIRST AND SECOND LINE.

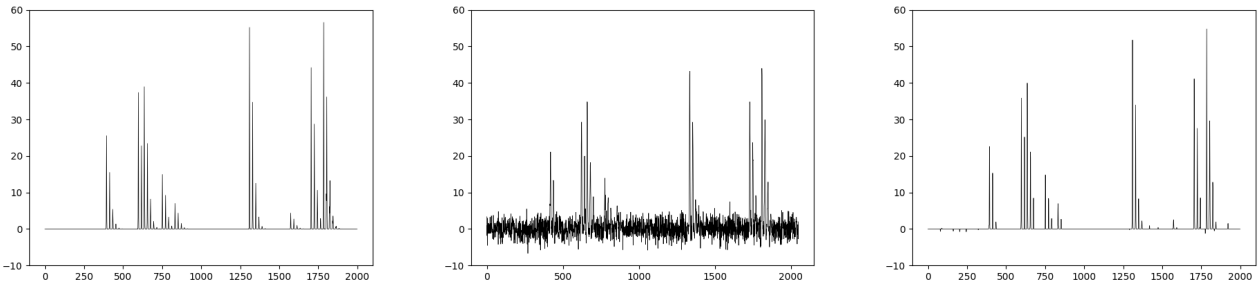

Fig. 5. Example of original signal $\bar{x}$, associated degraded observation z and reconstruction estimation using the proposed unrolled primal-dual approach.

V. CONCLUSIONS

This work explores deep unrolling approaches for sparse inverse problems. We performed the unrolling of a constrained primal-dual algorithm onto neural network architectures. Our experiments showcase the efficiency of our approach on two realistic datasets, with respect to several competing methods. Mainly, our method yields improved reconstruction quality with respect to iterative based methods and is competitive with respect to other unrolled methods.

REFERENCES

[1] H. Trussell and M. Civanlar, "The feasible solution in signal restoration," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 201–212, 1984.

[2] J. Honerkamp and J. Weese, "Tikhonovs regularization method for ill-posed problems," *Continuum Mechanics and Thermodynamics*, vol. 2, no. 1, pp. 17–30, 1990.

[3] R. A. Willoughby, "Solutions of ill-posed problems (an tikhonov and vy arsenin)," *SIAM Review*, vol. 21, no. 2, p. 266, 1979.

[4] B. Kaltenbacher, A. Neubauer, and O. Scherzer, "Iterative regularization methods for nonlinear ill-posed problems," in *Iterative Regularization Methods for Nonlinear Ill-Posed Problems*. de Gruyter, 2008.

[5] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to deblur," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 7, pp. 1439–1451, 2015.

[6] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[7] M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 85–95, 2017.

[8] D. Ren, W. Zuo, D. Zhang, L. Zhang, and M.-H. Yang, "Simultaneous fidelity and regularization learning for image restoration," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 284–299, 2019.

[9] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proceedings of the 27th international conference on international conference on machine learning*, 2010, pp. 399–406.

[10] C. Bertocchi, E. Chouzenoux, M.-C. Corbineau, J.-C. Pesquet, and M. Prato, "Deep unfolding of a proximal interior point method for image restoration," *Inverse Problems*, vol. 36, no. 3, p. 034005, 2020.

[11] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3929–3938.

[12] Y. Huang, E. Chouzenoux, and J.-C. Pesquet, "Unrolled variational bayesian algorithm for image blind deconvolution," *arXiv preprint arXiv:2110.07202*, 2021.

[13] M. Savanier, E. Chouzenoux, J.-C. Pesquet, and C. Riddell, "Deep unfolding of the dbfb algorithm with application to roi ct imaging with limited angular density," *arXiv preprint arXiv:2209.13264*, 2022.

[14] N. Komodakis and J.-C. Pesquet, "Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 31–54, 2014.

[15] J. Cheng, H. Wang, L. Ying, and D. Liang, "Model learning: Primal dual networks for fast mr imaging," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 21–29.

[16] J. Adler and O. Öktem, "Learned primal-dual reconstruction," *IEEE transactions on medical imaging*, vol. 37, no. 6, pp. 1322–1332, 2018.

[17] M. Jiu and N. Pustelnik, "A deep primal-dual proximal network for image restoration," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 2, pp. 190–203, 2021.

[18] X. Zhang, Q. Lian, Y. Yang, and Y. Su, "A deep unrolling network inspired by total variation for compressed sensing mri," *Digital Signal Processing*, vol. 107, p. 102856, 2020.

[19] H. H. Bauschke and P. L. Combettes, *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, 2011, vol. 408.

[20] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of mathematical imaging and vision*, vol. 40, no. 1, pp. 120–145, 2011.

[21] L. Condat, "A primal–dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *Journal of Optimization Theory and Applications*, vol. 158, pp. 460–479, 2013.

[22] C. Molinari, M. Massias, L. Rosasco, and S. Villa, "Iterative regularization for low complexity regularizers," *arXiv preprint arXiv:2202.00420*, 2022.

[23] P. L. Combettes and J.-C. Pesquet, "Fixed point strategies in data science," *IEEE Transactions on Signal Processing*, vol. 69, pp. 3878–3905, 2021.

[24] M. D. Zeiler, M. Ranzato, R. Monga, M. Mao, K. Yang, Q. V. Le, P. Nguyen, A. Senior, V. Vanhoucke, J. Dean *et al.*, "On rectified linear units for speech processing," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 3517–3521.

[25] L. M. Heaney, D. J. Jones, and T. Suzuki, "Mass spectrometry in medicine: a technology for the future?" p. FSO213, 2017.
[26] M. W. Senko, S. C. Beu, and F. W. McLaffertycor, "Determination of monoisotopic masses and ion populations for large biomolecules from resolved isotopic distributions," *Journal of the American Society for Mass Spectrometry*, vol. 6, no. 4, pp. 229–233, 1995.
[27] S. Mallat, *A wavelet tour of signal processing*. Elsevier, 1999.
[28] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 693–696.
[29] M. Gharbi, E. Chouzenoux, J.-C. Pesquet, and L. Duval, "Gpu-based implementations of mm algorithms. application to spectroscopy signal restoration," in *Proceedings of the 29th European Signal Processing Conference (EUSIPCO 2021)*, 2021, pp. 2094–2098.
[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.