MAP-informed Unrolled Algorithms for Hyper-parameter Estimation

Pascal Nguyen, Emmanuel Soubies, Caroline Chaux

MAP-INFORMED UNROLLED ALGORITHMS FOR HYPER-PARAMETER ESTIMATION

Pascal Nguyen[1]*, Emmanuel Soubies*[2]*, Caroline Chaux*[3]

[1]CNRS@CREATE, Singapore
[2]CNRS, IRIT, Univ Toulouse, Toulouse, France
[3]CNRS, IPAL, Singapore

ABSTRACT

Hyper-parameter tuning, and especially regularisation parameter estimation, is a challenging but essential task when solving inverse problems. The solution is obtained here through the minimization of a functional composed of a data fidelity term and a regularization term. Those terms are balanced through a (or several) regularisation parameter(s) whose estimation is made under an unrolled strategy together with the inverse problem solving. The resulting network is trained while incorporating information on the model through Maximum a Posteriori estimation which drastically decreases the amount of data needed for the training and results in better estimation results. The performances are demonstrated in a deconvolution context where the regularisation is performed in the wavelet domain.

Index Terms— Maximum a Posteriori, Unrolling, Parameter estimation, Deconvolution, Wavelets.

1. INTRODUCTION

In this work, we consider the class of inverse problems that consists in recovering $\mathbf{x} \in \mathbb{R}^N$ from data $\mathbf{y} \in \mathbb{R}^M$ that follow the linear model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\varepsilon}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{M\times N}$ is the forward matrix and $\boldsymbol{\varepsilon} \in \mathbb{R}^M$ a noise vector whose entries are drawn from a zero-mean normal distribution with variance σ^2. The standard practice to tackle such an inverse problem [1] is to solve an optimization problem of the form

$$\hat{\mathbf{x}} \in \left\{ \arg\min_{\mathbf{x}\in\mathbb{R}^N} \mathcal{F}(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \mathcal{R}(\mathbf{x}) \right\}. \tag{2}$$

Here, the least-squares term measures the discrepancy between the model and the data. While other measures of fit could be adopted, the ℓ_2-norm is a natural choice for additive Gaussian noise (see Section 2). The regularization term $\mathcal{R}$ (assumed convex) enforces prior knowledge on the targeted solution. Finally, the regularization parameter $\lambda > 0$ allows to adjust the trade-off between data fidelity and regularization.

The choice of an optimal value for λ is by no means straightforward and usually practitioners resort to manual tuning. Yet, given its practical importance, many works have been and continue to be devoted to the development of methods that select λ automatically. These include classical approaches such as cross-validation or L-curve [2], as well as more sophisticated methods like bi-level strategies [3]. Moreover, with the recent rise of neural networks and increasing computational capabilities, several methods based on deep learning were proposed [4]. In particular, some of them exploit unrolling strategies [5] so as to maintain interpretability.

This work was supported by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) Programme.

Contributions and outline. In this communication, we propose a method to automatically adjust λ from the data $\mathbf{y}$. To that end, we train—in an end-to-end supervised way—a network that combines a trainable parameter estimation module together with an unrolled algorithm for (2) (Section 3). As opposed to [5], the proposed parameter estimation module derives from a maximum a posteriori (MAP) interpretation of (2) (Section 2). As such, it remains interpretable and only few parameters have to be learned.

Although the proposed general principle (sections 2 and 3.1) can be adapted to any problem of the form (2), we focus in this work on wavelet-based deconvolution. It corresponds to the situation where

$$\mathbf{A} = \mathbf{H}\mathbf{W}^* \text{ and } \mathcal{R} = \|\cdot\|_1, \tag{3}$$

with $\mathbf{H} \in \mathbb{R}^{N\times N}$ being a convolution operator and $\mathbf{W} \in \mathbb{R}^{N\times N}$ a wavelet operator. Hence, in this case, $\mathbf{x}$ represents wavelets coefficients of the target image $\mathbf{z} = \mathbf{W}^*\mathbf{x}$. Moreover, we present in Section 3.4 an extension allowing for the consideration of a different λ for each wavelet sub-band.

We illustrate the effectiveness of the proposed method on image deconvolution purposes in Section 4. We show that 1) the network being informed, the proposed strategy enables to automatically estimate the regularisation parameter(s) from a small learning data set and 2) the reached performances are very close to the best performances one can obtain following an (unrealistic) grid search strategy. Indeed, the latter requires a ground truth and is not applicable in practice but constitutes a good reference for comparison.

2. MAXIMUM A POSTERIORI INTERPRETATION

From a Bayesian perspective, solving an inverse problem of the from (1) consists in maximizing $\mathbf{x} \mapsto p(\mathbf{x}|\mathbf{y})$, the probability of $\mathbf{x}$ knowing the data $\mathbf{y}$. Although not directly accessible in practice, Bayes' formula gives us

$$p(\mathbf{x}|\mathbf{y}) \propto p_\sigma(\mathbf{y}|\mathbf{x}) p_\mu(\mathbf{x}) \tag{4}$$

where $p_\sigma(\mathbf{y}|\mathbf{x})$ is the likelihood function which only depends on the noise level σ. For additive Gaussian noise, we have $p_\sigma(\mathbf{y}|\mathbf{x}) \propto \exp\left(-\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2/(2\sigma^2)\right)$ which is nothing else than the probability of the vector of noise $\boldsymbol{\varepsilon}$. On the other hand, p_μ represents the prior distribution on $\mathbf{x}$. Without loss of generality, we consider log-concave Gibbs distributions of the form $p_\mu(\mathbf{x}) \propto \exp\left(-\mathcal{R}(\mathbf{x})/\mu\right)$ where $\mu > 0$ is a scale parameter.

Injecting these expressions in (4) and taking the negative logarithm, we see that maximizing $p(\mathbf{x}|\mathbf{y})$ with respect to $\mathbf{x}$ is equivalent

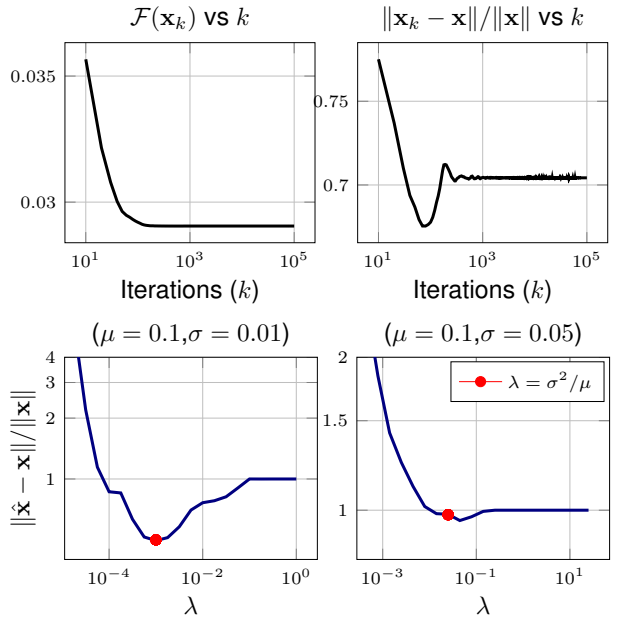

Fig. 1. **Top**: Typical evolution of the loss function of (2) and the relative error along with the iterations. Note the importance of attaining the true convergence regime where the relative error stabilizes. **Bottom**: Relative error of the solution of (2) as a function of λ. Both graphs have been obtained with $N = 300$, $\mathbf{A}$ a convolution operator with a Gaussian kernel, $\mathbf{x}$ generated from a zero-mean Laplace distribution with scale μ, and $\boldsymbol{\varepsilon}$ generated from a zero-mean normal distribution with variance σ^2. The left and right graphs correspond respectively to low and high noise regimes. The red point corresponds to the value $\lambda = \sigma^2/\mu$.

to solve (2) with

$$\lambda = \sigma^2/\mu. \tag{5}$$

As such, we get from this Bayesian viewpoint that the hyper-parameter λ should be proportional to the noise variance and inversely proportional to the scale of the prior distribution. This is confirmed by Fig. 1 where we illustrate, on synthetic data satisfying the model perfectly, that the smallest relative error is obtain by taking λ equal or close to the theoretical value defined in (5). In Section 3, we exploit this interpretation in order to derive a MAP-informed unrolled algorithm that allows to automatically adjust λ from the data $\mathbf{y}$.

3. PROPOSED METHODOLOGY

3.1. General Principle

Let $\hat{\sigma}_{\mathbf{y}}$ and $\hat{\mu}_{\mathbf{y}}$ be respectively estimates of σ and μ obtained from the data $\mathbf{y}$ (see Section 3.2). Then, one can deploy an iterative optimization algorithm to solve (2) with $\lambda = \hat{\sigma}_{\mathbf{y}}^2/\hat{\mu}_{\mathbf{y}}$. Yet, the success of this approach depends heavily on the quality of these estimates. To tackle this drawback, we propose to learn rectification functions $r_\sigma(\cdot;\boldsymbol{\theta})$ and $r_\mu(\cdot;\boldsymbol{\vartheta})$ (with learnable parameters $\boldsymbol{\theta}$ and $\boldsymbol{\vartheta}$) so that $r_\sigma(\hat{\sigma}_{\mathbf{y}};\boldsymbol{\theta})$ and $r_\mu(\hat{\mu}_{\mathbf{y}};\boldsymbol{\vartheta})$ lead to better estimates of σ and μ. Specifically, we consider rectification functions of the form

$$r_\sigma(s;\boldsymbol{\theta}) = \theta_1 s + \theta_2 \;\text{ and }\; r_\mu(u;\boldsymbol{\vartheta}) = \vartheta_1 u + \vartheta_2. \tag{6}$$

The rationale behind this choice is discussed in Section 3.2.

We then learn the four parameters defining these rectification functions through the resolution of

$$(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\vartheta}}) \in \arg\min_{\boldsymbol{\theta},\boldsymbol{\vartheta}\in\mathbb{R}^2} \sum_{q=1}^{Q} \|\mathcal{N}(\boldsymbol{\theta},\boldsymbol{\vartheta};\mathbf{y}_{\text{train}}^q) - \mathbf{x}_{\text{train}}^q\|_2^2 \tag{7}$$

where $\{\mathbf{x}_{\text{train}}^q, \mathbf{y}_{\text{train}}^q\}_{q=1}^Q$ is a set of input-target image-pairs and $\mathcal{N}$ is defined by unrolling an algorithm for (2) (see Section 3.3). More precisely, we set

$$\mathcal{N}(\boldsymbol{\theta},\boldsymbol{\vartheta};\mathbf{y}) = S^T(\mathbf{x}^0, (r_\sigma(\hat{\sigma}_{\mathbf{y}};\boldsymbol{\theta}))^2/r_\mu(\hat{\mu}_{\mathbf{y}};\boldsymbol{\vartheta})) \tag{8}$$

where $S^T(\mathbf{x}^0,\lambda)$ stands for T iterations of the considered algorithm initialized with $\mathbf{x}^0$ to solve (2) with the given λ.

By construction, the proposed network can adapt the hyper-parameter to both the noise level and the image content, as in the recent work [5]. Yet, as opposed to [5], our parameter estimation module is interpretable and has significantly less parameters to learn.

3.2. Initial Estimation of σ and μ

Initial estimation of σ. Following [6], we compute an initial estimate of σ as

$$\hat{\sigma}_{\mathbf{y}} = \frac{1}{0.6745}\text{median}(|d(\mathbf{W}\mathbf{y})|), \tag{9}$$

where d is a function that extracts detail coefficients of the wavelet decomposition. As illustrated in Fig. 2 (top), this estimator is very accurate in our context where $\mathbf{A} = \mathbf{H}\mathbf{W}$ is a low-pass filter. We observe that $\hat{\sigma}_{\mathbf{y}}$ detaches from the identity line only for very small level of noise ($\sigma < 10^{-5}$) compared to the signal level $\mu = 0.01$.

Initial estimation of μ. Given that the random vectors $\mathbf{A}\mathbf{x}$ and $\boldsymbol{\varepsilon}$ are independent, we have $\text{var}(\mathbf{y}) = \text{var}(\mathbf{A}\mathbf{x}) + \text{var}(\boldsymbol{\varepsilon})$ where var stands for the variance. In the Bayesian context of Section 2, setting $\mathcal{R} = \|\cdot\|_1$ corresponds to the consideration of a zero mean Laplace distribution with scale parameter μ [7]. As such, if $\mathbf{A}$ was an identity operator we would get

$$\text{var}(\mathbf{y}) = 2\mu^2 + \sigma^2, \tag{10}$$

using the fact that the variance of a zero mean Laplace distribution with scale parameter μ is $2\mu^2$. Although this relation does not hold anymore for an arbitrary $\mathbf{A}$, we use it as a rough approximation in order to compute an initial estimate of μ as

$$\hat{\mu}_{\mathbf{y}} = \sqrt{|\text{var}(\mathbf{y}) - \hat{\sigma}_{\mathbf{y}}^2|/2}. \tag{11}$$

The performance of this estimator is analyzed in the bottom graph of Fig. 2. We can distinguish two regimes. On the one hand, when the signal level (i.e. μ) is larger than the noise level (here $\sigma = 0.1$), the estimation is accurate within a constant bias. This is due to the fact that we ignore the effect of the operator $\mathbf{A}$. This bias motivates the proposed parameterization for the rectification functions in (6). On the other hand, when μ is smaller than the noise level, the estimation becomes constant, equal to a value related to the noise only.

Remark 1. *It is noteworthy to mention that we could directly adjust the parameters of the rectification functions in* (6) *from the experiments reported in Fig. 2. Yet, the proposed training strategy is more relevant for several reasons. First, natural images does not follow exactly the considered model (wavelet coefficients from a Laplace*

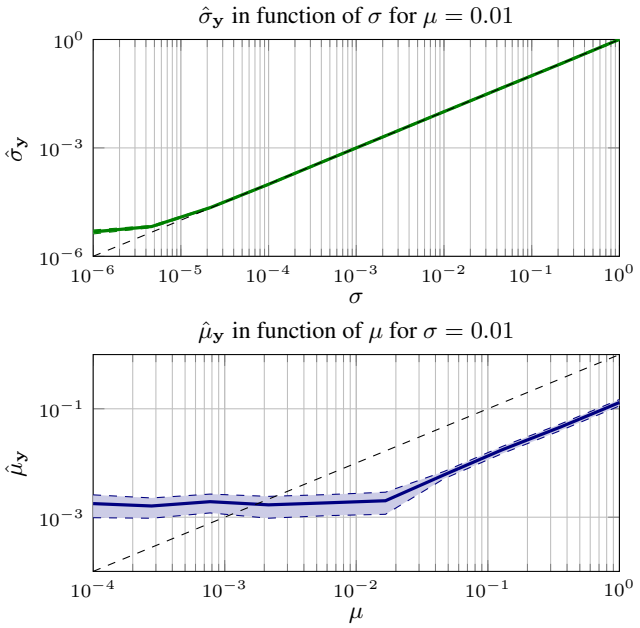

Fig. 2. **Performance of the estimators** (9) **(top) and** (11) **(bottom).** Mean (and standard deviation) of the estimators when $N = 1000$, $\mathbf{A}$ is a convolution operator with a Gaussian kernel, $\mathbf{x}$ is generated from a zero-mean Laplace distribution with scale μ, and $\boldsymbol{\varepsilon}$ is generated from a zero-mean normal distribution with variance σ^2. The closer the estimation is to the identity line, the better it is.

distribution). Second, as emphasized with the experiment of Fig. 1, $\lambda = \sigma^2/\mu$ is "optimal" when comparing solutions at convergence. The later being very slow (problem ill-conditioned), algorithms are usually stopped way before convergence. In this case, $\lambda = \sigma^2/\mu$ may not remain the best choice. Hence, the proposed training strategy in Section 3.1 allows to adjust the rectification functions by taking into account both the deviation of natural images from the considered model and a reduced number of algorithm iterations.

3.3. Unrolled Fast Iterative Soft Thresholding Algorithm (FISTA)

The resulting optimisation problem (2) is solved by using an unrolled [8] version of FISTA algorithm [9]. This algorithm basically involves two main operations: a gradient step (quadratic term in (2), stepsize γ) and a thresholding step (regularisation $\mathcal{R}$). In this case, T iterations of FISTA, denoted by $S^T(\mathbf{x}^0, \lambda)$ in (8), reads

Algorithm 1 FISTA

1: **Input:** $\mathbf{x}^0$, $\mathbf{v}^0 = \mathbf{x}^0$, $\lambda \geq 0$, $0 < \gamma < 1/\|\mathbf{A}^*\mathbf{A}\|$
2: **Output:** $\mathbf{x}^T$
3: **for** $t = 0, \ldots, T-1$ **do**
4: $\quad \mathbf{x}^{t+1} = \mathrm{Soft}_{\lambda\gamma}\left(\mathbf{v}^t - \gamma\mathbf{A}^*(\mathbf{A}\mathbf{x}^t - \mathbf{y})\right)$
5: $\quad \mathbf{v}^{t+1} = \mathbf{x}^t + \dfrac{t}{t+3}(\mathbf{x}^{t+1} - x^t)$
6: **end for**

where $\mathrm{Soft}_{\lambda\gamma}$ denotes the soft-thresholding operator defined by $\mathrm{Soft}_{\lambda\gamma}(\mathbf{x}) = \mathrm{sign}(\mathbf{x})\max(|\mathbf{x}| - \lambda\gamma, 0)$. This algorithm is implemented under an unrolling strategy that is under a neural network form where each layer is defined by one iteration of Alg. 1.

3.4. Wavelet Sub-band Variable Parameter

When considering a multiscale representation of signals, choosing an adaptive regularisation parameter is often more accurate, the information contained in each sub-band possibly being of a great variability [10]. We can thus generalize the optimisation problem (2) with a new one given by

$$\hat{\mathbf{x}} \in \left\{ \arg\min_{\mathbf{x}\in\mathbb{R}^N} \mathcal{G}(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \sum_{j\in\mathcal{S}_j} \lambda_j \mathcal{R}(\mathbf{x}_{|_j}) \right\}. \quad (12)$$

where $\mathcal{S}_j$ denotes the j^{th} sub-band of the multiscale transform, and $\mathbf{x}_{|_j}$ denotes the coefficients in $\mathbf{x}$ associated to this j^{th} sub-band. This formulation allows us to define (and later tune) a regularisation parameter λ_j per sub-band. Only Line 4 of Alg. 1 needs to be modified with a processing per sub-band.

It is noteworthy to mention that, in this new configuration, a grid search is no longer possible (in addition to being not realistic as mentioned previously) due to the increase of the number of parameters to tune. In contrast, the proposed MAP-informed unrolled strategy can be defined as described previously with $\mathcal{N}(\boldsymbol{\theta}, (\boldsymbol{\vartheta}_j)_j; \mathbf{y})$ where now a vector $\boldsymbol{\vartheta}_j$ per sub-band will be learnt.

4. NUMERICAL EXPERIMENTS

4.1. Context

We illustrate the performance of our approach in an image deconvolution context where the direct model is given by (1) in which $\mathbf{A} = \mathbf{H}\mathbf{W}^*$ and the noise $\boldsymbol{\varepsilon}$ corresponds to an additive white Gaussian noise with variance σ^2. $\mathbf{H}$ represents a blur operator corresponding to a Gaussian kernel (with $\sigma_h = 1$) and $\mathbf{W}^*$ (resp. $\mathbf{W}$) defines an orthogonal wavelet synthesis (resp. analysis) operator (Daubechies wavelet of order 4 on 3 resolution levels).

Our objective is to recover $\mathbf{x}$ from $\mathbf{y}$ assuming $\mathbf{H}$ is known by solving either Problem (2) (one regularisation parameter) or Problem (12) (multiple regularisation parameters).

We perform our training and tests on Linnaeus 5 Image database[1] where we considered images of size 256×256 with 256 gray-scale levels. We consider 50 unrolled iterations of FISTA. To learn the parameters of our rectification functions, we use 20 epochs of ADAM optimizer with learning rate 0.01 on 30 images taking into account 2 different levels of noise (two values of σ are randomly chosen between 1 and 15 for each of the 30 images). As such, a total of 60 images with various noise levels has been considered for training.

We test our procedure on 100 images corrupted by two different intensities of noise: a low noise level (standard deviation 2) and a high noise level (standard deviation 10).

4.2. Results

To assess the relevance of the proposed approach, we compare the performances obtained by an exhaustive grid search and the unrolled strategies with single and multiple λ (in the trained and untrained contexts). First, we display the average Signal-To-Noise Ratio (SNR) performances (over the 100 tested images) for the four unrolled strategies in Fig. 4. As expected, we observe that the trained network always outperforms its untrained counterpart. The multi λ strategy is always as good as its single counterpart (performances depend on the considered image).

[1] http://chaladze.com/l5/

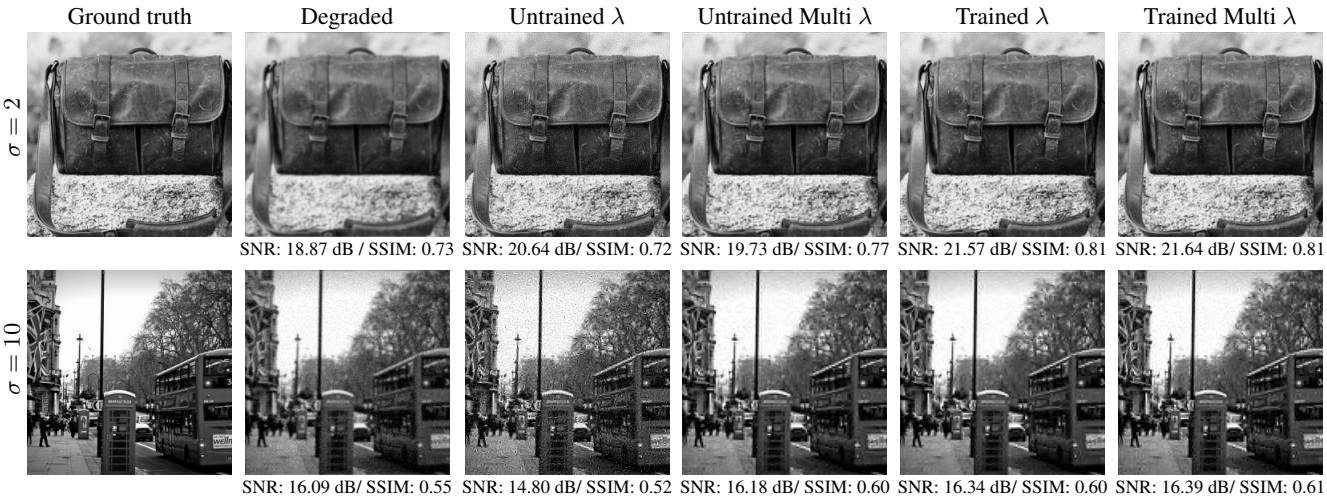

Fig. 3. **Visual performances.** Low noise level (top) and high noise level (bottom).

We now concentrate our attention on two images of the test dataset (represented in Fig. 3 upper and lower left corners) for which we display the SNR performances along with λ in Fig. 5. Not only does the trained network always outperforms its untrained counterpart, but the trained versions (with slightly improved performance for the multi λ case) reach the maximum performance of the single λ grid search. Associated visual performance are displayed in Fig. 3.

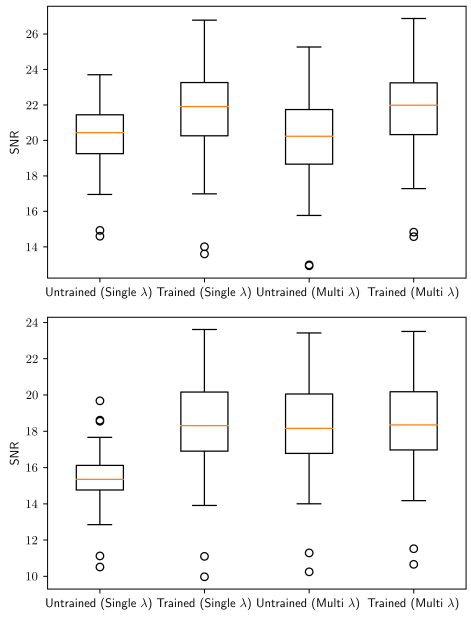

Fig. 4. **Average numerical performances over the dataset.** Low noise level (top) and high noise level (bottom).

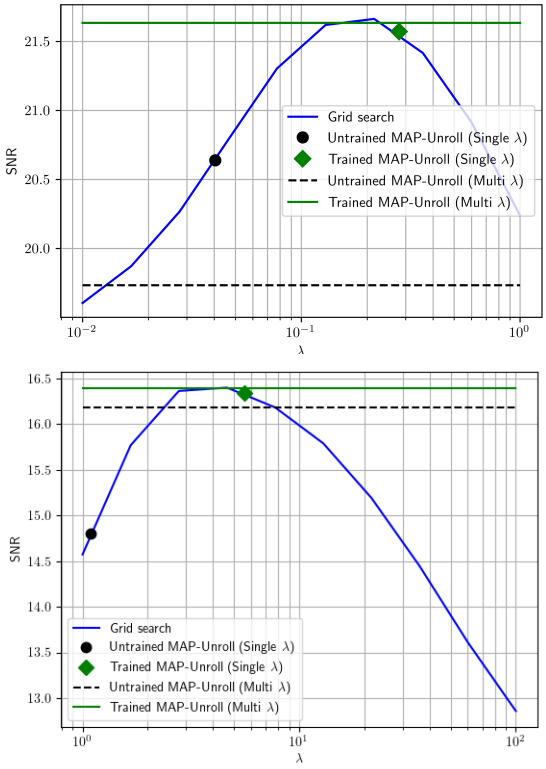

Fig. 5. **SNR performances along with λ.** Low noise level (top) and high noise level (bottom).

5. CONCLUSION

We have proposed in this work a MAP-informed unroll procedure that allows to adjust automatically the regularisation parameter when solving inverse problems in a regularized variational framework. The information provided to the network comes from the MAP principle at a low cost and enables the training step to be performed on small datasets (small number of parameters to be learnt). Furthermore, the use of an unrolled neural network allows to keep the interpretability of the whole process. Finally, it is noteworthy to mention that this strategy goes beyond the considered Problem (2) and can be extended to other optimisation problems.

6. REFERENCES

[1] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs, "A variational formulation for frame based inverse problems," *Inverse Problems*, vol. 23, no. 4, pp. 1495–1518, Aug. 2007.

[2] A. Cultrera and L. Callegaro, "A simple algorithm to find the L-curve corner in the regularisation of ill-posed inverse problems," *IOP SciNotes*, vol. 1, no. 2, pp. 025004, 8 2020.

[3] C. Crockett and J. Fessler, "Bilevel methods for image reconstruction," *Found. Trends Signal Process.*, vol. 15, no. 2-3, pp. 121–289, 2022.

[4] B. Afkham, J. Chung, and M. Chung, "Learning regularization parameters of inverse problems via deep neural networks," *Inverse Problems*, vol. 37, no. 10, pp. 105017, 9 2021.

[5] A. Kofler, F. Altekrüger, F. A. Ba, C. Kolbitsch, E. Papoutsellis, D. Schote, C. Sirotenko, F. F. Zimmermann, and K. Papafitsoros, "Learning regularization parameter-maps for variational image reconstruction using deep neural networks and algorithm unrolling," *arXiv preprint arXiv:2301.05888*, 2023.

[6] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[7] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, 1 1996.

[8] V. Monga, Y. Li, and Y. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Process. Mag.*, vol. 38, no. 2, pp. 18–44, 3 2021.

[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[10] S. Chang, Y. Bin, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Process.*, vol. 9, no. 9, pp. 1532–1546, 2000.