FIRMNET: A SPARSITY AMPLIFIED DEEP NETWORK FOR SOLVING LINEAR INVERSE PROBLEMS

Praveen Kumar Pokala†*, Amol G Mahurkar*†*, and Chandra Sekhar Seelamantula*

Department of Electrical Engineering, Indian Institute of Science, Bangalore-12, India
praveenkumar@iisc.ac.in, amolgmahurkar@gmail.com, chandra.sekhar@ieee.org

ABSTRACT

Recovering a sparse signal from a noisy linear measurement is an important problem in signal processing. Typically, one employs greedy pursuit techniques such as OMP, CoSaMP to solve an ℓ_0 regularization problem. For large-scale problems, iterative shrinkage techniques such as ISTA, FISTA, AMP-ℓ_1 have been introduced. The underlying formulation in the iterative algorithms is a LASSO problem with an ℓ_1-penalty. It is known in the literature that an ℓ_1-penalty in LASSO suffers from underestimation of large signal amplitudes. Also, the iterative shrinkage-based approaches such as ISTA typically have only one free parameter to trade-off between noise variance and sparsity. We consider a minimax-concave penalty-based formulation, which offers an unbiased estimate of the sparse signal. The resulting iterative firm-thresholding algorithm is restructured as a DNN architecture called *FirmNet*. The proposed network, *FirmNet*, has two interpretable shrinkage function parameters – one that controls the noise variance, and the other that allows for explicit sparsity control. We compare the network with a broader network architecture of Learned-ISTA (LISTA), and show that it outperforms in terms of the probability-of-error-in-support (PES) – a strong support recovery metric, by at least three-fold. We also observe an improvement of 2 to 4 dB in reconstruction SNR compared with LISTA.

***Index Terms*—** Sparse coding, iterative algorithms, proximal operators, deep neural networks, deep unfolding.

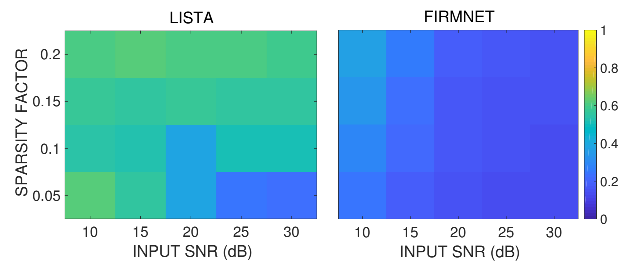
1. INTRODUCTION

Sparse coding is the problem of recovering a sparse vector $\mathbf{x}^\star \in \mathbf{R}^n$ from a compressed $(m < n)$ noisy linear measurement vector $\mathbf{y} \in \mathbf{R}^m$. The linear measurement process is given by $\mathbf{y} = D\mathbf{x}^\star + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is additive i.i.d. Gaussian noise. An estimate of $\mathbf{x}^\star$ requires solving an ℓ_0 problem. Greedy pursuit techniques [1–4] such as OMP [5, 6], CoSaMP [7], Subspace Pursuit [8] are used to estimate $\mathbf{x}^\star$ for a given sparsity level. Another family of techniques involves a convex relaxation to the ℓ_0 penalty, namely, the ℓ_1 norm resulting in the LASSO optimization problem:

$$\arg\min_{\mathbf{x}} \ \frac{1}{2}\|\mathbf{y} - D\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_1. \tag{1}$$

When the problem dimensionality is small, an interior-point algorithm [9] is employed. For large-scale problems, one has to rely on iterative shrinkage algorithms, such as Iterative Shrinkage/Thresholding Algorithm (ISTA) [10], a faster version of ISTA (FISTA) [11] that incorporates the momentum factor. Recently, Donoho et al. proposed the Approximate Message Passing (AMP) [12, 13] algorithm to solve the problem in (1) as a special case.

† Equal contribution.



(a) (b)

Fig. 1: [color online] This plot highlights the robustness of FirmNet to LISTA in Probability of Error in Support (PES) [15] metric for over 1000 test realizations. The LISTA estimate has a larger support than intended as indicated by a high PES in contrast to much lower PES scores of FirmNet. Lower the PES, superior the performance.

The AMP algorithm incorporates a term proportional to the past residuals called an *Onsager correction* into one of the update steps of ISTA. Addition of the correction term decouples the AMP iterations such that the input to the shrinkage is an additive white Gaussian noise (AWGN) corrupted version of the true signal $\mathbf{x}^\star$ with a known variance [14]. A variant of AMP for solving the LASSO problem is called AMP-ℓ_1 [13]. To counter the fragility of the AMP-ℓ_1 algorithm to the matrix D, Rangan et al. introduced *Vector AMP* (VAMP) [14]. The VAMP algorithm while keeping the desirable properties of AMP, such as fewer iterations for convergence and shrinkage inputs that follow the AWGN model (as described above) over a larger classes of matrices, which are large and right-rotationally invariant [14]. In conclusion, the above three main classes of algorithms (ISTA, AMP, and VAMP) that are aimed at solving (1) rely on shrinkage functions with a free parameter and a fixed linear transformation.

LeCun et al. restructured the way sparse coding problems are solved by proposing a deep neural network (DNN)-like architecture, which is learnable as well as interpretable [16]. An iterative soft-thresholding algorithm is *unfolded* into a network to learn the affine transformation matrices and MSE-optimal shrinkage functions. The update steps in the ISTA formulation are cascaded over several iterations to form a layered network called learned ISTA (LISTA) [16]. This approach enhances the robustness over ISTA by achieving lower normalized MSE (NMSE). Recently, several existing iterative algorithms got revamped into a learnable network framework and have offered superior performance over their iterative counterparts. The message passing algorithms (AMP and VAMP) got revamped into a *Learnable AMP* (LAMP) and a *Learnable VAMP* (LVAMP) DNN architectures [14]. The architecture of LAMP is similar to that of LISTA except the additional "bypass" paths incorporating the *On-*

sager correction. Ghanem et al. proposed ISTA-Net and ISTA-Net$^+$ by taking an approach different from LISTA by learning the parameters, such as nonlinear transforms, shrinkage thresholds, step-sizes, etc., end-to-end in a joint formulation [17]. Mahapatra et al. modeled the nonlinear transformation as a linear expansion of thresholds (LET) [18], and restructured ISTA and FISTA into corresponding DNN architectures *LETnet* and *fLETnet*, respectively [19]. These networks have shown an improvement of 3 to 4 dB in terms of reconstruction SNR.

The approaches in the past decade have focused on solving an ℓ_1 problem [1, 2, 4] in (1) employing either iterative algorithms [10, 12, 14, 20–22] or DNN-like architectures [14, 16, 17, 19, 23, 24]. It is known in the literature that an ℓ_1 penalty suffers from underestimation of large amplitude values inducing a bias in the estimated sparse code, thereby fundamentally limiting its performance [25]. Also, except for signals with a high SNR and low sparsity factor ρ, an ℓ_1 penalty tends to shrink signal amplitudes and has a larger support than intended. This phenomenon can be observed in Figure 1a, where a higher PES score (defined in (14)) indicates a larger support of an estimate. We show that the proposed network architecture, FirmNet, has a lower PES indicating accurate estimation of the support. We propose a new iterative algorithm and its equivalent deep neural network (DNN) architecture, which ameliorates the underestimation in large amplitude values and promotes super-sparsity [26] compared with other iterative ℓ_1-based algorithms.

Of particular relevance to the current work is the contribution of Voronin and Woerdeman [21], who proposed an iterative varied thresholding algorithm (IVTA), in order to counter the effects of underestimation of large amplitudes while maintaining the desired sparsity. To generalize the two-step procedure in [27] consisting two soft-thresholding operations, they proposed a direct approach utilizing firm-thresholding operators. Since the firm-thresholding operator leads to a non-convex penalty making the overall cost non-convex, the IVTA algorithm starts with soft-thresholding steps, and gradually switches to firm-thresholding as the sparsity increases to recover the underestimated amplitudes. This heuristic approach though shown to be faster and accurate for some denoising applications is not robust when the estimated sparsity level required to initiate the iterations is not accurate.

A class of non-convex penalties proposed in [25] generalizes ℓ_1 norm while maintaining the convexity of the least-squares cost function to be minimized. The proposed generalized minimax-concave (MC) penalty $\psi_B(\mathbf{x})$ is the difference between an ℓ_1 norm and a generalized Huber function $S_B(\mathbf{x})$. Selesnick provides a convexity condition (in (47) of [25]) that the scaling matrix B should satisfy. The MC penalty bridges the gap between an ℓ_1 norm and an ℓ_p norm ($p < 1$). The advantage of the MC penalty over ℓ_1 is that it provides a more accurate estimation of high-amplitudes and promotes higher level of sparsity in the estimate. In the following section, inspired by Selesnick's recent contribution, we formulate a locally convex cost with a non-convex MC penalty ensuring that the condition in *Proposition 3* of [25], i.e., a positive curvature of the cost, is satisfied.

An illustration that compares LISTA [16], the first generation of interpretable DNNs, and the proposed DNN architecture, FirmNet, using a strong support recovery metric called the probability of error in support (PES) [15] is shown in Figure 1. The proposed DNN architecture, FirmNet, is shown to outperform LISTA by at least three-fold in estimating the correct support. In this paper, we consider the reformulation of LASSO to promote *super-sparsity* in the solution. To this extent, we consider the use of minimax-concave (MC) penalty replacing the ℓ_1 norm [25]. The MC penalty weighs amplitudes above a threshold with a fixed weight, thereby eliminating the underestimation issue and promotes super-sparsity by virtue of non-convexity under the threshold value. The main contributions of this paper are as follows:

1. An iterative firm-thresholding algorithm (IFTA) is proposed to solve the reformulated MC penalty-based problem. We employ the proximal operator for the non-convex penalty, which has an additional free parameter (for sparsity control) compared to the shrinkage parameter in ISTA. This proposition overcomes the inherent trade-off between the support recovery and amplitude underestimation.
2. Finally, we propose an IFTA-inspired DNN architecture called FirmNet to learn the Huber-optimal shrinkage parameters and affine transformations. We learn two shrinkage parameters, one that controls the noise variance and the other that controls sparsity in the solution. The Huber loss is preferred to mean-squared error (MSE) to impart robustness to outliers.

This paper is organized as follows. We first reformulate the LASSO problem with a sparsity-promoting non-convex penalty [28]. We then present in Section 2.1 the iterative firm-thresholding algorithm based on the proximal operator derived from the non-convex penalty [29, 30]. Further, in Section 2.2, we restructure the iterative algorithm and unfold it into a DNN-like architecture. In Section 3, we show the efficacy of the proposed FirmNet, and compare it to a broader network architecture of LISTA. We would like to emphasize that comparison with LISTA is sufficient, since the underlying formulation of DNN-like architectures is a LASSO formulation.

2. PROBLEM FORMULATION

Consider the signal model $\mathbf{y} = D\mathbf{x}^\star + \boldsymbol{\epsilon}$, where $\mathbf{y} \in \mathbb{R}^m$ is the noisy measurement, $D \in \mathbb{R}^{m\times n}$ is an overcomplete ($m < n$) dictionary, $\mathbf{x}^\star \in \mathbb{R}^n$ is a sparse vector and $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is the additive measurement noise. To overcome the bias introduced by the ℓ_1 penalty in LASSO, we introduce the minimax-concave (MC) penalty [25] as a sparsity-promoting regularizer:

$$\arg\min_{\mathbf{x}} \underbrace{\|\mathbf{y} - D\mathbf{x}\|_2^2}_{f(\mathbf{x})} + g_\gamma(\mathbf{x}; \lambda), \tag{2}$$

where

$$g_\gamma(x_i; \lambda) = \begin{cases} \lambda\left(|x_i| - \dfrac{x_i^2}{2\gamma\lambda}\right), & |x_i| \le \gamma\lambda, \\ \dfrac{\gamma\lambda^2}{2}, & |x_i| \ge \lambda\gamma, \end{cases} \tag{3}$$

for all $i \in [\![1, n]\!]$ and

$$g_\gamma(\mathbf{x}; \lambda) = \sum_{i=1}^{n} g_\gamma(x_i; \lambda).$$

In the next section, we solve the optimization problem in (2) using the Majorization-Minimization (MM) [31] approach. We refer to the resulting algorithm as an Iterative Firm Thresholding Algorithm (IFTA).

2.1. Iterative Firm Thresholding Algorithm (IFTA)

We now describe an iterative algorithm for solving (2). The quadratic term $f(\mathbf{x})$ in (2) is expanded using a first-order approximation at point $\mathbf{x}^k$ as given below:

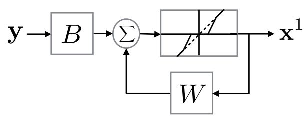


Fig. 2: An IFTA iteration, which resembles a layer in DNN.

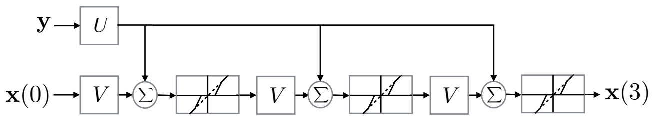


Fig. 3: An illustration of a 3-layered FirmNet. The matrices U, V, and the nonlinearities $g_\gamma(\mathbf{x};\lambda)$ are learned.

$$f_k(\mathbf{x}) \approx f(\mathbf{x}^k) + (\mathbf{x} - \mathbf{x}^k)^{\mathrm{T}} \nabla f(\mathbf{x}^k). \tag{4}$$

The update $\mathbf{x}^{k+1}$ using the MM approach is given as

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} f_k(\mathbf{x}) + \frac{1}{2\eta}\|\mathbf{x} - \mathbf{x}^k\|_2^2 + g_\gamma(\mathbf{x};\lambda), \tag{5}$$

which can be expressed equivalently as

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \frac{1}{2\eta}\left\|\mathbf{x} - (\mathbf{x}^k - \eta \nabla f(\mathbf{x}^k))\right\|_2^2 + g_\gamma(\mathbf{x};\lambda). \tag{6}$$

The proximal operator corresponding to the penalty $g_\gamma(\,\cdot\,;\lambda)$ is

$$F_\gamma^g(\mathbf{u};\lambda) = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{u}\|_2^2 + g_\gamma(\mathbf{x};\lambda). \tag{7}$$

Since $g_\gamma(\mathbf{x};\lambda)$ is separable in x, the resultant firm-shrinkage operator $F_\gamma^g(\mathbf{u};\lambda)$ is given as

$$F_\gamma^g(u_i;\lambda) = \begin{cases} 0, & |u_i| \leq \lambda, \\ \dfrac{\gamma}{\gamma - 1}\ (|u_i| - \lambda)\,\mathrm{sign}(u_i), & \lambda < |u_i| \leq \gamma\lambda, \\ u_i, & |u_i| > \gamma\lambda, \end{cases} \tag{8}$$

where $u_i = [\mathbf{u}]_i\,, \forall i \in [\![1, n]\!]$. Solving (6) is equivalent to

$$x_i^{k+1} = \arg\min_{x_i} \frac{1}{2\eta}\left(x_i - u_i^k\right)^2 + g_\gamma(x_i;\lambda), \tag{9}$$

where $u_i^k = \left[\mathbf{u}^k\right]_i, \forall i \in [\![1, n]\!]$ and $\mathbf{u}^k = \mathbf{x}^k - \eta\nabla f(\mathbf{x}^k)$.

The optimization problem in (7) is convex for $\gamma > 1$ [28]. The IFTA algorithm for the optimization problem in (9) with firm-thresholding operator is described in Algorithm 1. An update step of IFTA algorithm at $(k+1)^{\text{th}}$ iteration is $\mathbf{x}^{k+1} = F_\gamma^g(W\mathbf{x}^k + B\mathbf{y}\,;\lambda\eta)$, where $W = (I - \eta D^{\mathrm{T}} D)$ and $B = \eta D^{\mathrm{T}}$. Here, $\mathbf{x}^{k+1}$ can also be interpreted as applying an affine transformation on $\{\mathbf{x}^k, \mathbf{y}\}$, followed by a nonlinearity, which is the firm-thresholding operator parameterized by γ, λ. Each iteration of IFTA can be seen as a layer in a neural network as shown in Figure 2. In the next section, we unfold this iterative firm-thresholding algorithm (IFTA) into a DNN-like architecture (FirmNet), and learn the network parameters from training data.

2.2. FirmNet: IFTA-Inspired DNN

As seen in the previous section, an iteration of IFTA resembles one layer in a DNN architecture. We further extend the idea to learn the network parameters $\{U, V, \lambda, \gamma\}$ instead of fixed weights $\{B, W, \lambda, \gamma\}$.

Algorithm 1: Iterative Firm Thresholding Algorithm (IFTA).

Data: $\mathbf{y}, D, \eta = 1/\|D\|_2^2, k_{\max}$
Initialization: $\mathbf{x}^0, k = 0$
while $k \leq k_{max}$ **do**
  $\mathbf{z}^{k+1} = \mathbf{x}^k - \eta g^k$ where $g^k = -D^{\mathrm{T}}(\mathbf{y} - D\mathbf{x}^k)$
  $\mathbf{x}^{k+1} = F_\gamma^g(\mathbf{z}^{k+1};\lambda\eta)$
  $k = k + 1$
end
Result: $\hat{\mathbf{x}}_0 \leftarrow \mathbf{x}^k$

We propose a DNN architecture called FirmNet with a fixed depth and each layer structured as shown in Figure 2. The sparsity control parameter γ and shrinkage parameter λ (which reduces noise in an estimate) gives us better interpretability of the unfolded network. An example of a three-layered architecture is shown in Figure 3.

The estimated output of FirmNet is denoted by $\tilde{\mathbf{x}} = z(\mathbf{y}, \theta)$, where $\theta = \{U, V, \lambda, \gamma\}$ are all trainable parameters. We use stochastic gradient-descent (SGD) to minimize the loss function $C(\theta)$, averaged over the training data $\{\mathbf{y}_i, \mathbf{x}_i^\star\}_{i=1}^M$ and defined as follows

$$C(\theta) = \frac{1}{M}\sum_{i=1}^{M} L_\delta(\mathbf{x}_i^\star, z(\mathbf{y}_i, \theta)), \quad \text{s.t.} \quad \lambda > 0, \gamma > 1, \tag{10}$$

where, for a user-defined parameter δ,

$$L_\delta(\mathbf{x}, z(\mathbf{y},\theta)) = \begin{cases} \dfrac{1}{2}\|\mathbf{x} - z(\mathbf{y},\theta)\|_2^2, & \|\mathbf{x} - z(\mathbf{y},\theta)\|_1 \leq \delta, \\ \delta\|\mathbf{x} - z(\mathbf{y},\theta)\|_1 - \dfrac{1}{2}\delta^2, & \text{otherwise.} \end{cases} \tag{11}$$

This architecture can be viewed as a recurrent neural network (RNN), unfolded over iterations (synonymous with time).

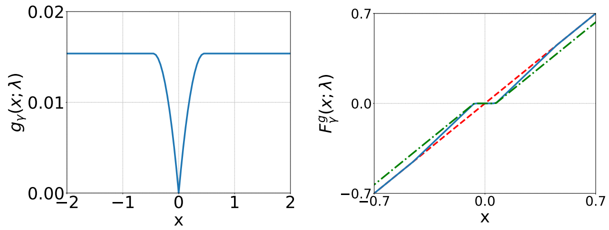
3. EXPERIMENTS AND DISCUSSION

In this section, we evaluate the performance of the proposed DNN architecture, FirmNet, over the more general, broader network architecture of Learned ISTA (LISTA) [16]. We use 15 layers in FirmNet as well as LISTA for comparison. In the experiments, the entries of an overcomplete dictionary $D \in \mathbf{R}^{m\times n}$ are drawn from an i.i.d. normal distribution, $\mathcal{N}(0, m^{-1})$. The entries of $\mathbf{x}^\star \in \mathbf{R}^n$ are drawn from an i.i.d. $\mathcal{N}(0, 1)$ with probability ρ (i.e., $\mathbf{x}^\star$ is Bernoulli-Gaussian). We refer to ρ as the sparsity factor. The undersampling factor $\gamma_c = m/n \in (0, 1)$. During training, we initialize the parameters $\{U, V, \mathbf{x}(0)\}$ of the FirmNet and LISTA with $\{\eta D^{\mathrm{T}}, (I - \eta D^{\mathrm{T}} D), \mathbf{0}\}$ for a fair comparison.

We consider the objective measures such as reconstruction signal-to-noise (R-SNR) and probability of error in support (PES) [15] to quantify the performance. Here, R-SNR is evaluated over 1000 test realizations. The probability of error in support is a strong support recovery metric and is evaluated over 1000 test realizations. Reconstruction SNR (R-SNR) between the predicted sparse code $\tilde{\mathbf{x}}$ and the true sparse code $\mathbf{x}^\star$ is defined as

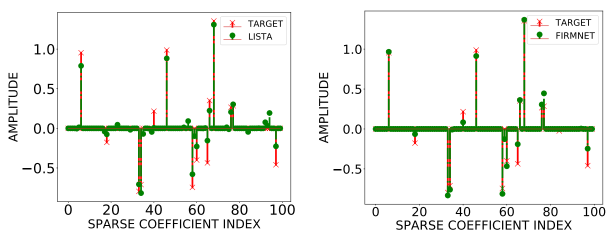
$$\text{R-SNR} = 10\log_{10}\left(\frac{\|\mathbf{x}^\star\|_2^2}{\|\tilde{\mathbf{x}} - \mathbf{x}^\star\|_2^2}\right)\text{dB}. \tag{12}$$

In Equation (3.29) of [15], the measure to quantify the effec-



(a) (b)

Fig. 4: [color online] The learned (a) parametric MC penalty $g_\gamma(\mathbf{x};\lambda)$ with its corresponding (b) firm-thresholding operator $F_\gamma^g(\mathbf{x};\lambda)$ (blue). The optimal parameters for input SNR of 20 dB were $\gamma = 6.68$ and $\lambda = 0.06$. The original signal $y = x$ line (red), and a soft-thresholding operator (green) highlights the bias in estimation.



(a) (b)

Fig. 5: [color online] Recovered sparse code from LISTA and the proposed FirmNet with target vector for 20 dB input SNR and undersampling factor γ_c of 0.7. The superiority of FirmNet over LISTA in terms of correct estimation of support and amplitude values can be seen in this plot.

tiveness of a sparse recovery algorithm in estimating true support is given as

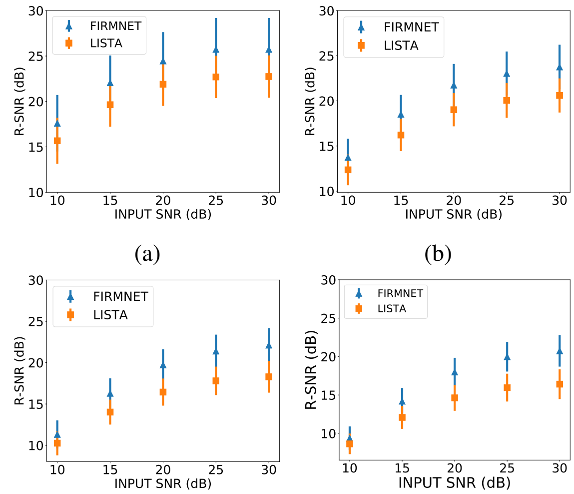
$$d(\mathcal{S}(\tilde{\mathbf{x}}), \mathcal{S}(\mathbf{x}^\star)) = \frac{\max(|\mathcal{S}(\tilde{\mathbf{x}})|, |\mathcal{S}(\mathbf{x}^\star)|) - |\mathcal{S}(\tilde{\mathbf{x}}) \cap \mathcal{S}(\mathbf{x}^\star)|}{\max(|\mathcal{S}(\tilde{\mathbf{x}})|, |\mathcal{S}(\mathbf{x}^\star)|)}, \tag{13}$$

where $|\cdot|$ is the cardinality of the argument and $\mathcal{S}(\mathbf{x})$ is the support of $\mathbf{x}$. The probabililty of error in support (PES) is defined as

$$\text{PES} = \frac{1}{N}\sum_{i=1}^{N} d(\mathcal{S}(\tilde{\mathbf{x}}_i), \mathcal{S}(\mathbf{x}_i^\star)). \tag{14}$$

We observe that the bias an ℓ_1 penalty (LISTA) induces is contained by the MC penalty (FirmNet) in Figure 4(b). The aforementioned bias (in their corresponding proximal operator) is rendered as an underestimation of the signal amplitude. This in turn causes the R-SNR for LISTA to drop as evident in Figure 6.

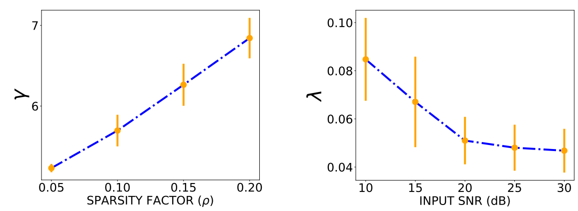
Another shortcoming of the ℓ_1 penalty is that it can not differentiate a shrinkage parameter λ for shrinking noise in the measurement and sparsity of the estimate. As the input SNR decreases or sparsity factor increases, the shrinkage parameter λ in LISTA doesn't offer a good trade-off with one free parameter λ. With high noise, λ is lower, assigning more weight to $f(\mathbf{x})$ in (2), leading to a dense estimate. This translates to a higher PES as seen in Figure 1(a). The proposed network, FirmNet, has an additional free parameter γ exclusively for sparsity control. This parameter allow us to induce sparsity in an estimate while maintaining a separate control over noise variance $f(\mathbf{x})$. We recover nearly exact support with a lower PES score



(a) (b)

(c) (d)

Fig. 6: [color online] Consistent improvement of FirmNet's R-SNR compared to LISTA's over input SNRs is shown for sparsity factor (a) $\rho = 0.05$, (b) $\rho = 0.1$, (c) $\rho = 0.15$, and (d) $\rho = 0.2$. ($\gamma_c = 0.7$)



(a) (b)

Fig. 7: [color online] These plots highlight the role of the learned parameter γ in controlling the sparsity ρ. Also, the role of the learned parameter λ, which varies with input SNR, underscores the control over noise variance.

and high R-SNR as shown in Figure 1(b) and Figure 6, respectively.

Reformulating the iterative algorithms into a DNN-like architecture makes the network interpretable. In FirmNet, there are two free parameters (γ and λ) parameterizing the nonlinearity in addition to the affine transformations U and V. In Figure 7, we affirm that the additional parameter γ (of the MC penalty) indeed controls the sparsity, as it increases almost linearly with the sparsity level ρ. Also, the shrinkage parameter λ corresponds to the noise variance, and decreases drastically with increasing input SNR.

4. CONCLUSION

In this paper, we reconsidered fundamentally the penalty used (i.e., an ℓ_1-norm in LASSO) to recover a true sparse vector $\mathbf{x}^\star$. We showed that the MC penalty would result in a nearly unbiased estimate. An iterative firm-thresholding algorithm (IFTA) was proposed to solve the optimization problem with a non-convex MC penalty using a firm-thresholding proximal operator. We further demonstrated the efficacy of an unrolled IFTA-inspired DNN architecture called FirmNet, and showed that it outperforms LISTA, in terms of probability of error in support [15] – a strong support recovery metric, by at least three-fold. The proposed network also offers an additional degree of control for sparsity, and has precise interpretability for sparsity level and noise variance.

5. REFERENCES

[1] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[2] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[4] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007.

[5] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[6] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, pp. 40–44, 1993.

[7] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[8] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[10] I. Daubechies, M. I. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[11] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 693–696, 2009.

[12] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18914–18919, 2009.

[13] A. Montanari, "Graphical models concepts in compressed sensing," *Compressed Sensing: Theory and Applications*, pp. 394–438, 2012.

[14] M. Borgerding, P. Schniter, and S. Rangan, "AMP-inspired deep networks for sparse linear inverse problems," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4293–4308, 2017.

[15] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer Publishing Company, Incorporated, 1st edition, 2010.

[16] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proceedings of the 27th Intl. Conf. on Machine Learning*, pp. 399–406, 2010.

[17] J. Zhang and B. Ghanem, "ISTA-Net: Iterative shrinkage-thresholding algorithm inspired deep network for image compressive sensing," June 2018, arXiv:1706.07929v2 [Online]. Available: https://arxiv.org/abs/1706.07929v2.

[18] H. Pan and T. Blu, "An iterative linear expansion of thresholds for ℓ_1-based image restoration," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp. 3715–3728, 2013.

[19] D. Mahapatra, S. Mukherjee, and C. S. Seelamantula, "Deep sparse coding using optimized linear expansion of thresholds," May 2017, arXiv:1705.07290 [Online]. Available: https://arxiv.org/abs/1705.07290.

[20] J. Liang and C.-B. Schönlieb, "Faster FISTA," July 2018, arXiv:1807.04005 [Online]. Available: https://arxiv.org/abs/1807.04005.

[21] S. Voronin and H. J. Woerdeman, "A new iterative firm-thresholding algorithm for inverse problems with sparsity constraints," *Applied and Computational Harmonic Analysis*, vol. 35, no. 1, pp. 151–164, 2013.

[22] J. Ma and L. Ping, "Orthogonal AMP," *IEEE Access*, vol. 5, pp. 2020–2033, 2017.

[23] C. Metzler, A. Mousavi, and R. Baraniuk, "Learned D-AMP: Principled neural network based compressive image recovery," in *Advances in Neural Information Processing Systems*, pp. 1772–1783, 2017.

[24] S. Mukherjee, D. Mahapatra, and C. S. Seelamantula, "DNNs for sparse coding and dictionary learning," NIPS Bayesian Deep Learning Workshop, 2017, Available: http://bayesiandeeplearning.org/2017/papers/31.pdf.

[25] I. Selesnick, "Sparse regularization via convex analysis," *IEEE Transactions on Signal Processing*, vol. 65, no. 17, pp. 4481–4494, 2017.

[26] I. Selesnick, "Sparsity amplified," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4356–4360, 2017.

[27] I. Loris, G. Nolet, I. Daubechies, and F. A. Dahlen, "Tomographic inversion using ℓ_1-norm regularization of wavelet coefficients," *Geophysical J. Intl.*, vol. 170, no. 1, pp. 359–370, 2007.

[28] C.-H. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *The Annals of Statistics*, vol. 38, no. 2, pp. 894–942, 2010.

[29] U. S. Kamilov and H. Mansour, "Learning optimal nonlinearities for iterative thresholding algorithms," *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 747–751, 2016.

[30] J. Woodworth and R. Chartrand, "Compressed sensing recovery via nonconvex shrinkage penalties," *Inverse Problems*, vol. 32, no. 7, pp. 075004, 2016.

[31] M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, "Majorization–minimization algorithms for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2980–2991, 2007.