GRAPH SIGNAL RECOVERY FROM INCOMPLETE AND NOISY INFORMATION USING APPROXIMATE MESSAGE PASSING

Gita Babazadeh Eslamlou[1], *Alexander Jung*[2], *Norbert Goertz*[1] *and Mehdi Fereydooni*[3]

[1]Institute of Telecommunications, Vienna University of Technology, Austria; {gita.babazadeh, norbert.goertz}@nt.tuwien.ac.at
[2]Dept. of Computer Science, Aalto University, Finland; alexander.jung@aalto.fi
[3]Dept. of Computer Eng and IT, Amirkabir University of Technology, Iran; fereydooni@aut.ac.ir

ABSTRACT

We consider the problem of recovering a graph signal from noisy and incomplete information. In particular, we propose an approximate message passing based iterative method for graph signal recovery. The recovery of the graph signal is based on noisy signal values at a small number of randomly selected nodes. Our approach exploits the smoothness of typical graph signals occurring in many applications, such as wireless sensor networks or social network analysis. The graph signals are smooth in the sense that neighboring nodes have similar signal values. Methodologically, our algorithm is a new instance of the denoising based approximate message passing framework introduced recently by Metzler et. al. We validate the performance of the proposed recovery method via numerical experiments. In certain scenarios our algorithm outperforms existing methods.

Index Terms— Graph signal denoising, compressed sensing, approximate message passing, subsampling.

1. INTRODUCTION

A recent approach to deal with large-scale datasets occurring in big data applications such as genetics, image processing and social network analysis is the theory of graph signal processing [1]. By considering discrete time signals as being defined over a chain graph (the nodes represent the time instants) graph signal processing is obtained by allowing for general graphs as the signal domain. E.g., a graph signal occurring in a wireless sensor network is defined over a graph whose nodes represent the sensors and the edges model connectivity between those sensors. Another example for graph signals is given by 2D images where each pixel corresponds to a node in a grid graph connecting the nearest pixels with each other. A main goal of graph signal processing is to derive sampling theorems which are important for design of graph signal processing systems. By paralleling the Nyquist-Shannon theory for band-limited signals, the authors of [2] define notions of (approximately) band-limitedness for graph signals. For a given bandwidth they also construct sampling sets of minimum size which guarantee perfect recovery. It is also possible to define the notion of sparse graph signals and apply concepts of compressed sensing (CS) to graph-structured signals. A first application of CS to graph signals is based on the graph Fourier transform (GFT) which is composed of the eigenvectors of the graph Laplacian. Assuming that the graph signal is sparse in the GFT domain, the authors of [3] propose to subsample the graph signal according to CS theory. However, in this paper we consider a notion of sparsity different from [3]. In particular, we assume that the graph signal of interest is sparse in the sense of consisting of few clusters within which the signal is approximately constant. This notion of graph signal sparsity can be interpreted as a constraint on the total variation of the graph signal. Thus, our goal is to reconstruct graph signals under a total variation constraint. For discrete time signals, total variation based denoising has been considered in [4], which applies the approximate message passing (AMP) framework for denoising structured signals. Moreover, the authors of [5] present a widely applicable framework, termed denoising-based approximate message passing (DAMP). This framework is based on combining a given denoiser functions, tailored to a specific signal model, with the AMP rationale of iteratively recovering signals from incomplete random measurements. However, to the best of our knowledge, the use of the DAMP framework for graph signal denoising employing a total-variation constraint is novel.

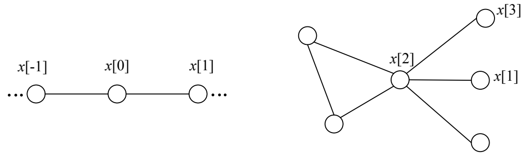
Contributions: Our main contribution is an AMP based graph signal recovery method which is able to cope with incomplete and noisy measurements. This method can be regarded as an instance of the denoising framework in [5] for the special case of graph signals having small total variation, i.e., which consist of few clusters within which the signal values do not vary significantly. The signal recovery is based on a small number of noisy samples of the smooth graph signal. We conduct illustrative numerical experiments which validate the performance of the proposed recovery method and reveal superiority of our approach against existing methods in some relevant scenarios.

Outline: The rest of the paper is organized as follows. We formalize the problem of graph signal recovery from incomplete and noisy measurements in Section 2. The novel DAMP based recovery method is presented in Section 3. Finally, some numerical results are presented and discussed in Section 4.

2. PROBLEM SETUP

2.1. Elements of Graph Signal Processing

The emerging field of graph signal processing [1, 6] aims at dealing efficiently with decentralized, graph-structured data



(a) discrete time signal (b) generic graph signal

Fig. 1. (a) Chain graph underlying discrete time signal processing and (b) generic graph signal.

as encountered in modern information networks. Graph signal processing is a generalization of discrete time signal processing. Specifically, discrete time signals may be interpreted as graph signals define over a chain graph whose nodes represent the discrete time instants. A general graph signal is obtained by allowing for general graph structures (see Fig. 1). More formally, we consider undirected weighted graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$ with node set $\mathcal{V} = \{1, \ldots, N\}$ and the edge set $\mathcal{E}$ consisting of unordered node pairs (r, s) for which $W_{r,s} \neq 0$. For a given graph $\mathcal{G}$, a graph signal $\mathbf{x}$ is a mapping from the node set into the reals. We can represent a graph signal conveniently as a vector $\mathbf{x} \in \mathbb{R}^N$ by defining x_r to be the value of the graph signal at node $r \in \mathcal{V}$.

We consider graph signals that are defined over undirected and weighted graphs. A non-zero entry of the weight matrix $W_{r,s}$ represents the strength of the correlation between signal values x_r and x_s. In a wireless sensor network application, the entry $W_{r,s}$ could reflect the distance between sensor nodes r and s. It is reasonable to assume the sensor values x_r and x_s of nearby sensor nodes to be strongly correlated. Another important matrix associated with a graph is the graph Laplacian matrix $\mathbf{L}$, defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{W}. \tag{1}$$

Here, $\mathbf{D}$ denotes the diagonal matrix with the rth diagonal element given by $D_{r,r} = \sum_{r' \in \mathcal{V}} D_{r,r'}$, i.e., the sum of the weights of all the edges connected to node r.

Many methods of discrete time signal processing (e.g., denoising), rely on smoothness of the signal with respect to the underlying graph. In order to make the notion of graph signal smoothness precise, we follow [6] and introduce the graph gradient

$$\|\nabla_r \mathbf{x}\|_2 := \left[\sum_{r' \in \mathcal{N}(r)} W_{r,r'} (x_{r'} - x_r)^2 \right]^{1/2}. \tag{2}$$

The norm $\|\nabla_r \mathbf{x}\|$ of the local gradient is termed local variation and measures the variability of the graph signal at a given node r. Here, $\mathcal{N}(r) := \{r' : W_{r,r'} \neq 0\}$ is the neighborhood of node $r \in \mathcal{V}$. A global measure of the graph signal smoothness is then obtained by:

$$S_p(\mathbf{x}) = (1/p) \sum_{r \in \mathcal{V}} \|\nabla_r \mathbf{x}\|^p \tag{3}$$

for some $p \in [1, \infty)$. In what follows we will consider only two specific choices for p, i.e., $p = 1$ and $p = 2$. For $p = 1$ the measure $S_p(\mathbf{x})$ is termed the *graph total variation* [7] and when $p = 2$ the measure $S_p(\mathbf{x})$ reduces to the *graph Laplacian form* [6]

$$S_2(\mathbf{x}) = (1/2) \sum_{r \in \mathcal{V}} \sum_{r' \in \mathcal{N}(r)} W_{r,r'} (x_{r'} - x_r)^2 = \mathbf{x}^T \mathbf{L} \mathbf{x}. \tag{4}$$

We have now the tools at hand to formalize the graph signal recovery problem considered in this paper.

2.2. The Recovery Problem

Our approach is based on the hypothesis that the true graph signal $\mathbf{x}$ is smooth, i.e., the measure $S_p(\mathbf{x})$ ($p \in \{1, 2\}$) is small. We have access to the graph signal $\mathbf{x}$ only via its values at a randomly selected small subset $\mathcal{S} = \{i_1, \ldots, i_M\} \subseteq \mathcal{V}$ of graph nodes. Moreover, the observed signal values are corrupted by measurement noise. Thus, the observation is given by

$$\mathbf{y} = \mathbf{x}\big|_{\mathcal{S}} + \sigma \mathbf{n}, \tag{5}$$

where the restriction $\mathbf{x}\big|_{\mathcal{S}}$ is obtained from $\mathbf{x}$ by selecting the entries of $\mathbf{x}$ with indices in $\mathcal{S}$. The noise vector $\mathbf{n}$ is assumed to be white Gaussian noise with zero-mean and unit variance, i.e., $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Alternatively, we can represent the vector $\mathbf{y}$ as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \sigma \mathbf{n}. \tag{6}$$

The measurement matrix $\mathbf{A} \in \{0, 1\}^{M \times N}$ models the selection of the subset $\mathcal{S}$: It contains exactly one non-zero element in each row, i.e., $A_{r,i_r} = 1$. Since the subset $\mathcal{S}$ is chosen randomly, the matrix $\mathbf{A}$ is random as well.

While the recovery of graph signals from incomplete noisy measurements has been considered already in [8], the application of AMP to the recovery of smooth graph signals seems to be new. Since our recovery problem can be interpreted as a structured signal recovery problem using incomplete information, we will now propose a recovery method based on the DAMP framework which is well suited for such recovery problms.

3. GRAPH SIGNAL DENOISING VIA DAMP

3.1. Review of DAMP

Consider a signal $\mathbf{x} \in \mathbb{R}^N$ which is known to belong to some signal class $\mathcal{C}$, e.g., graph signals with small total variation. For many important signal classes $\mathcal{C}$, one can find efficient denoising functions $\mathbf{D}^{\sigma}(\cdot)$ which operate on the noisy signal

$$\mathbf{y} = \mathbf{x} + \sigma \mathbf{n} \tag{7}$$

where $\mathbf{n}$ is modeled as zero-mean white Gaussian noise with unit variance, i.e., $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The denoising mapping $\mathbf{D}^{\sigma}(\cdot)$ typically depends on the variance σ^2 of the additive noise in (7). However, the notation $\mathbf{D}^{\sigma}(\cdot)$ does not make explicit that the denoiser also depends on the signal model $\mathcal{C}$. The output $\mathbf{D}^{\sigma}(\mathbf{y})$ of the denoiser, when applied to the noisy signal $\mathbf{y}$, is an estimate for the true signal $\mathbf{x}$. For $\mathcal{C}$ being the set of sparse signals, i.e., $\mathcal{C} = \mathcal{X}_s := \{\mathbf{s} \in \mathbb{R}^N : \|\mathbf{s}\|_0 \leq s\}$ it is known that an efficient denoising mapping is obtained by retaining the s largest (magnitude) entries of $\mathbf{y}$ and zeroing the rest.

However, as opposed to the signal in noise model (7), recently much interest has been devoted to the problem of recovering a structured signal $\mathbf{x}$ from incomplete information given by noisy low-dimensional random projections

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \sigma\mathbf{n}, \tag{8}$$

where $\mathbf{A} \in \mathbb{R}^{M \times N}$, with $M \ll N$, denotes a random projection matrix. A widely used choice for $\mathbf{A}$ is the Gaussian ensemble which is a matrix consisting of i.i.d. zero-mean Gaussian variables with variance $1/M$ (which ensures column normalization). By leveraging the principles behind AMP, which considers the special case of sparse signals $\mathcal{C} = \mathcal{X}_s$, the authors of [5] propose an efficient iterative method, termed DAMP, for recovering a structured signal $\mathbf{x} \in \mathcal{C}$ from the measurements $\mathbf{y}$ in (8) for a general signal class $\mathcal{C}$ with associated denoiser $\mathbf{D}^{\sigma}(\cdot)$.

In particular, DAMP constructs a sequence $\hat{\mathbf{x}}_t$, $t = 1, 2, \ldots$, of signal estimates by iterating the following steps [5, Eq. (4)]

$$\hat{\mathbf{x}}_{t+1} = \mathbf{D}^{\hat{\sigma}_t}(\mathbf{A}^T\mathbf{z}_t + \hat{\mathbf{x}}_t) \tag{9}$$

$$\mathbf{z}_t = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_t + \frac{1}{M}\mathbf{z}_{t-1}\nabla \cdot \mathbf{D}^{\hat{\sigma}_{t-1}}(\mathbf{A}^T\mathbf{z}_{t-1} + \mathbf{x}_{t-1}) \tag{10}$$

$$\hat{\sigma}_t = \sqrt{(1/M)\|\mathbf{z}_t\|_2^2}. \tag{11}$$

The initial choice for the estimate $\hat{\mathbf{x}}$ and residual $\mathbf{z}$ is $\hat{\mathbf{x}}_0 = \mathbf{0}$ and $\hat{\mathbf{z}}_0 = \mathbf{y}$, respectively.

The DAMP iterations (9)–(11) are able to accurately recover the structured signal $\mathbf{x} \in \mathcal{C}$ from the noisy measurements $\mathbf{y}$ in (8). The correction term in (10), i.e., $\frac{1}{M}\mathbf{z}_{t-1}\nabla \cdot \mathbf{D}^{\hat{\sigma}_{t-1}}(\mathbf{A}^T\mathbf{z}_{t-1} + \mathbf{x}_{t-1})$ is crucial for the success of the DAMP algorithm. The effect of including this term in (10) is that the equivalent estimation noise $\mathbf{n}_t := \mathbf{A}^T\mathbf{z}_t + \hat{\mathbf{x}}_t - \mathbf{x}$ behaves nearly like a multivariate normal random vector [5]. The (approximate) Gaussianity of the effective noise vector $\mathbf{n}_t$ is clearly desirable since the denoiser $\mathbf{D}^{\hat{\sigma}_t}(\cdot)$ is typically trimmed to remove additive Gaussian noise and the first step (9) of DAMP just amounts to the denoising operation

$$\hat{\mathbf{x}}_{t+1} = \mathbf{D}^{\hat{\sigma}_t}(\mathbf{x} + \mathbf{n}_t). \tag{12}$$

3.2. Graph Signal DAMP (GSDAMP)

Let us now specialize the generic DAMP algorithm, given by the iterations (9)–(11), to the problem of graph signal denoising. We assume that the true signal belongs to the class $\mathcal{C}$ of smooth graph signals, given explicitly by $\mathcal{C} = \{\mathbf{x} : S_p(\mathbf{x}) \leq \rho\}$. A natural choice for the corresponding denoiser, which is to be applied to a noisy graph signal $\mathbf{s} = \mathbf{x} + \sigma\mathbf{n}$, would be given by the minimizer of the following problem:

$$\min_{\mathbf{x}' \in \mathbb{R}^N} \|\mathbf{s} - \mathbf{x}'\|_2^2 \quad \text{s.t.} \quad S_p(\mathbf{x}') \leq \rho. \tag{13}$$

However, we will find it more convenient to use the "penalized version" of (13), i.e.,

$$\mathbf{D}^{\sigma}(\mathbf{s}) := \underset{\mathbf{x}' \in \mathbb{R}^N}{\arg\min} \|\mathbf{s} - \mathbf{x}'\|_2^2 + \lambda S_p(\mathbf{x}'). \tag{14}$$

For convex $S_p(\mathbf{x}')$, which is the case for $p \geq 1$, the two problems (13) and (14) are equivalent by Lagrangian duality [9]. In fact, for each choice of ρ there exists a choice for λ such that a minimizer (13) is simultaneously a solution to (14) and vice-versa.

In order to deploy DAMP for graph signal denoising, we require an efficient implementation of the denoiser mapping $\mathbf{D}^{\sigma}(\cdot)$ and its divergence $\nabla \cdot \mathbf{D}^{\sigma}(\mathbf{x}) := \sum_{k=1}^{N} \frac{\partial}{\partial x_k}\mathbf{D}_k^{\sigma}(\mathbf{x})$. Note that the denoiser amounts to solving a convex optimization problem allowing for efficient numerical implementations. In particular, we will rely on the freely available software package GSPBox [10]. In order to evaluate the divergence $\nabla \cdot \mathbf{D}^{\sigma}(\mathbf{x})$, we follow [5]: An approximation of the divergence can be obtained by [11]

$$\nabla \cdot \mathbf{D}^{\sigma}(\mathbf{x}) \approx \mathrm{E}_{\mathbf{b}}\big\{(1/\varepsilon)\mathbf{b}^T(\mathbf{D}^{\sigma}(\mathbf{x} + \varepsilon\mathbf{b}) - \mathbf{D}^{\sigma}(\mathbf{x}))\big\} \tag{15}$$

for some small $\varepsilon > 0$. However, in the numerical implementation we will make a further approximation by replacing the expectation in (15) with a sample mean, i..e, we use

$$\tilde{d}(\mathbf{x}) := \frac{1}{L}\sum_{l=1}^{L}(1/\varepsilon)\mathbf{b}_l^T(\mathbf{D}^{\sigma}(\mathbf{x} + \varepsilon\mathbf{b}_l) - \mathbf{D}^{\sigma}(\mathbf{x})), \tag{16}$$

where $\mathbf{b}_1, \ldots, \mathbf{b}_L$ are i.i.d. realizations of the random vector $\mathbf{b} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

The summary of our method, which we term graph signal denoising using AMP (GSDAMP), is given in Alg.1.

Algorithm 1 (GSDAMP)**.** *Given the noisy graph signal samples* $\mathbf{y}$ *(cf.* (8)*), sampling pattern* $\mathcal{S}$*, and denoising parameter* p *(cf.* (14)*) perform the following:*

Step 1: *Initialize* $t = 0$, $\hat{\mathbf{x}}_0 = \mathbf{0}$, $\hat{\mathbf{z}}_0 = \mathbf{y}$,

Step 2: *implement a DAMP iteration via*

- $\tilde{\mathbf{x}}_t = \mathbf{A}^T\mathbf{z}_t + \hat{\mathbf{x}}_t$
- $\hat{\mathbf{x}}_{t+1} = \mathbf{D}^{\hat{\sigma}_t}(\tilde{\mathbf{x}}_t)$ *(using the denoiser* (14)*)*
- $\mathbf{z}_t = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_t + \frac{1}{M}\mathbf{z}_{t-1}\tilde{d}(\tilde{\mathbf{x}}_{t-1})$ *(using approximation* (16)*)*

Step 3: $t := t + 1$

Step 4: *If stoping criterion satisfied: output final estimate* $\hat{\mathbf{x}}_t$*, otherwise go back to Step 2.*

There are various possibilities for the stopping criterion in Step 4 of Alg. 1, e.g. a maximum number of iterations. For the numerical experiments discussed in Section 4, we used as convergence criterion the relative progress $\frac{\|\hat{\mathbf{x}}_t - \hat{\mathbf{x}}_{t-1}\|_2}{\|\hat{\mathbf{x}}_t\|_2}$ and stopped if it was below a given threshold ϵ.

4. NUMERICAL RESULTS

We present the results of the numerical experiments validating the performance of the proposed method. In particular, we analyze the normalized mean square error (NMSE) for

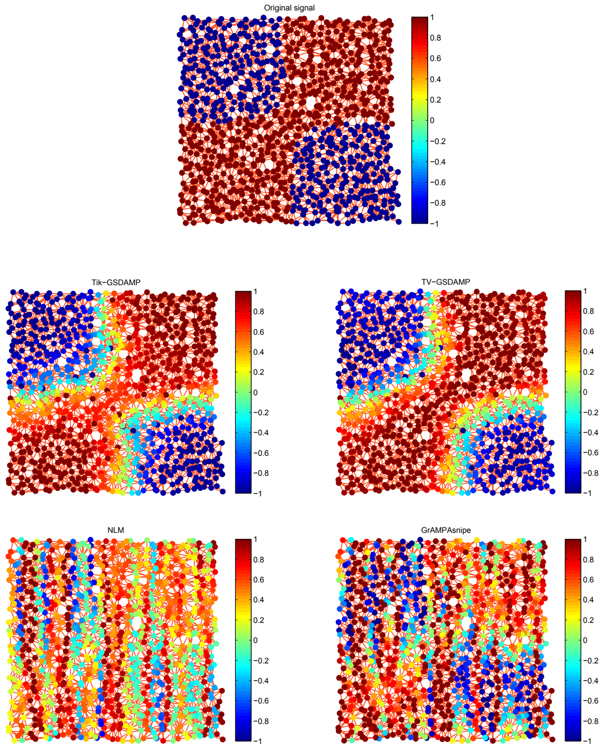


Fig. 2. The original graph $\mathcal{G}_0$ and its recovery with four different recovery algorithms for a sampling rate $M/N = 0.1$ and a noise variance $\sigma^2 = 0.1$.

varying sampling rate M/N, and noise variance σ^2. Here, we use Alg.1 with total variation ($p = 1$ in (3)) [7] and Tikhonov ($p = 2$ in (3)) [12] denoisers. We refer to these two instances of Alg.1 as TV-GSDAMP and Tik-GSDAMP respectively, and compare their efficiency with NLM [13] and Grampa [14] algorithms.

Using the GSPBox software [10], we generate an undirected graph $\mathcal{G}_0$, with size $N = 1000$. The graph signals are piece-wise smooth taking values 1 and -1 coded by red and blue colours in Fig. 2. In particular, Fig. 2 shows the original graph signal $\mathcal{G}_0$ and four recovered graph signals of the mentioned denoising algorithms. As evident, the graph signals recovered by TV-GSDAMP and Tik-GSDAMP are much more similar to the original graph signal than the results of NLM and Grampa.

The influence of the pairs $(M/N, \sigma^2)$ of sampling rate M/N and noise variance σ^2 on recovery performance is investigated in Fig. 3. The solid line shows the border between successful and failing reconstruction of the graph signal. In this experiment, the graph signal recovery is considered successful, when NMSE $\leqslant 0.28$ and the success region of each reconstruction algorithm is located below the curves. As Fig. 3 illustrates, both Tik-GSDAMP and TV-GSDAMP show significantly better recovery performance for low sampling rate, i.e., when $M/N < 0.4$.

In Fig. 4, we plot the NMSE over sampling rate M/N for TV-GSDAMP and Tik-GSDAMP. We show the corresponding plots for the NLM and Grampa recovery methods as well. As one can observe from Fig. 4, the NMSE of TV-GSDAMP and Tik-GSDAMP is smaller compared to other solvers, except for sampling rates larger than 0.8, where Grampa outperforms our algorithm (for both $p = 1$ and $p = 2$).

We also investigated the dependence of NMSE on the noise variance σ^2 for sampling rates of 0.3 and 0.5. As evident from Fig. 5, our algorithm is particularly superior at very low sampling rates.

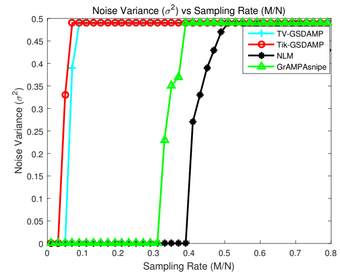


Fig. 3. The empirical comparison of the successful-recovery range of the investigated algorithms.

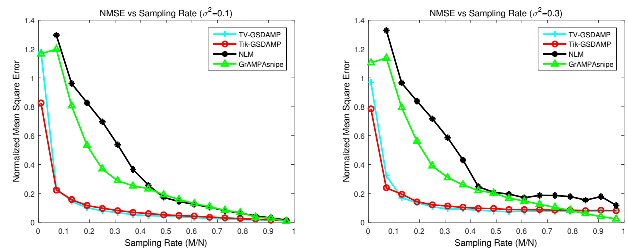


Fig. 4. NMSE vs sampling rate M/N, where the noise variance σ^2 is set to 0.1 and 0.3.

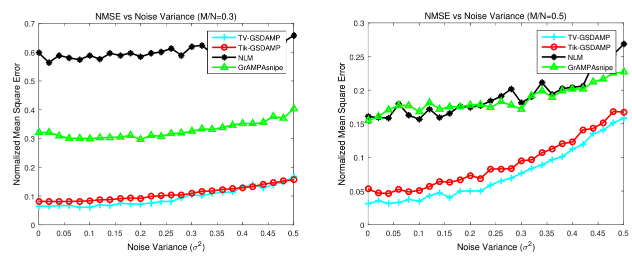


Fig. 5. NMSE vs noise variance σ^2, where the sampling rate M/N is set to 0.3 and 0.5.

5. CONCLUSION

We present a new AMP-based method for recovering smooth graph signals based on a small number of noisy signal samples. For the recovery, we combine graph signal denoisers with the AMP framework. In several relevant regimes, particularly for very low sampling rates, our method outperforms existing methods signifcantly.

REFERENCES

[1] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs: representation and processing of massive data sets with irregular structure," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, sep 2014.

[2] S. Chen, R. Varma, A. Singh, and J. Kovacevic, "Signal recovery on graphs: Random versus experimentally designed sampling," *International Conference on Sampling Theory and Applications (SampTA)*, pp. 337–341, may 2015.

[3] X. Zhu and M. Rabbat, "Graph spectral compressed sensing for sensor networks," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2865–2868, 2012.

[4] J. Kang, H. Jung, H.-N. Lee, and K. Kim, "Bernoulli-Gaussian approximate message-passing algorithm for compressed sensing with 1D-finite-difference sparsity," *arXiv preprint arXiv:1408.3930*, 2014.

[5] C. Metzler, A. Maleki, and R. Baraniuk, "From denoising to compressed sensing," *arXiv preprint arXiv:1406.4175*, no. 2, pp. 1–32, 2014.

[6] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[7] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 460–489, 2005.

[8] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal denoising on graphs via graph filtering," *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 872–876, dec 2014.

[9] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

[10] N. Perraudin, J. Paratte, D. Shuman, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "GSPBOX: A toolbox for signal processing on graphs," *arXiv preprint arXiv:1408.5781*, 2014.

[11] S. Ramani, T. Blu, and M. Unser, "Monte-Carlo Sure: A black-box optimization of regularization parameters for general denoising algorithms," *IEEE Transactions on Image Processing*, vol. 17, no. 9, pp. 1540–1554, sep 2008.

[12] G. H. Golub, P. C. Hansen, and D. P. O'Leary, "Tikhonov Regularization and Total Least Squares," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 1, pp. 185–194, jan 1999.

[13] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 60–65, 2005.

[14] M. Borgerding, P. Schniter, J. Vila, and S. Rangan, "Generalized approximate message passing for cosparse analysis compressive sensing," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3756–3760, apr 2015.