ABSTRACT

TAN, JIN. Fast and Robust Compressive Signal Reconstruction via Approximate Message Passing and Denoising. (Under the direction of Dror Baron.)

The rapid development of information technology in recent decades has led to an increasing demand for data acquisition systems, such as audio recorders and imaging devices. Moreover, large amounts of data must be stored or transmitted, and the time or power consumption required for transmission must be reduced. Although many types of data are high dimensional, they can often be represented efficiently when being projected to another space, which implies that the data is sparse or compressible in this projected space. Compressive sensing is an emerging field that employs the sparsity property of data and acquires the data in a compressive fashion. Our goal is to reconstruct the original data from those compressive measurements. Generalized approximate message passing (GAMP) and approximate message passing (AMP) are iterative compressive signal reconstruction algorithms that enjoy many mathematical and practical advantages. In this dissertation we utilize these advantages and develop fast and robust compressive signal reconstruction algorithms.

This dissertation contains two parts. In the first part, we explore compressive signal reconstruction algorithms that achieve application-oriented reconstruction quality. In order to evaluate the quality of reconstruction, standard error metrics such as square error are usually evaluated, and algorithms are developed to minimize these error metrics. However, reconstruction algorithms that minimize standard error metrics may not be robust, because in some applications they may not achieve satisfactory reconstruction quality. Therefore, we propose a compressive signal reconstruction algorithm that modifies the last iteration of GAMP and minimizes a broad range of user-defined error metrics, which provides great flexibility in achieving application-oriented performance.

Infinity norm error, also known as worst-case error, is an example of such non-standard error metrics. By minimizing the infinity norm error, we ensure that the reconstructed error for each signal component is modest. We explore some interesting theoretical properties of the worst-case error of signal reconstruction procedures.

In the second part of this dissertation, we study compressive imaging problems. Compressive imaging is an important application of compressive sensing, where the signals of interest are images. First, we consider a general compressive imaging problem where the imaging process is modeled by a random matrix; we then look into practical hyperspectral imaging problems, where the imaging process is modeled by a highly structured matrix that is far from random. Considering that AMP is an iterative signal reconstruction framework that performs scalar denoising at each iteration, we apply efficient and robust image denoisers within AMP iterations.

Numerical results show that our algorithms for random matrix imaging and hyperspectral imaging are both fast and robust, and improve over the state of the art in terms of runtime and reconstruction quality.

Fast and Robust Compressive Signal Reconstruction
via Approximate Message Passing and Denoising

by
Jin Tan

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina

2015

APPROVED BY:

Liyi Dai	Brian Hughes
Min Kang	Hamid Krim
Keith Townsend	Dror Baron Chair of Advisory Committee

DEDICATION

To my parents.

BIOGRAPHY

Jin Tan received the B.Sc. degree in microelectronics from Fudan University, China, in 2010, and the M.Sc. degree in electrical engineering from North Carolina State University, Raleigh, USA, in 2012. In 2011, she started working with Dr. Dror Baron as a Ph.D. student. Her research interests include information theory, statistical signal processing and estimation, and compressive image processing.

During the summer of 2014 and 2015, she worked with Siemens Corporate Research, Princeton, NJ and Cyberonics, Inc., Houston, TX, respectively, both as a research intern, where she applied her knowledge to medical image reconstruction and epilepsy seizure detection.

ACKNOWLEDGEMENTS

I would like to first thank my advisor Dr. Dror Baron for his kind and detailed instruction. From him, I gradually learned how to think as a researcher, present research results, and work independently. Besides being a supervisor, he is also a good friend, and helped me as a student studying abroad.

I would like to thank my committee members, in alphabetical order, Dr. Liyi Dai, Dr. Brian Hughes, Dr. Min Kang, Dr. Hamid Krim, and Dr. Keith Townsend. All their invaluable suggestions have been extremely helpful, and I am more than honored to have them as my committee members.

I would also like to thank my coauthors, in alphabetical order, Dr. Gonzalo Arce, Danielle Carmon, Dr. Liyi Dai, Nikhil Krishnan, Yanting Ma, and Hoover Rueda. It was a great experience collaborating with them. They helped me appreciate a broad range of research topics, and I would have been far less productive without their gracious help.

Besides Nikhil Krishnan and Yangting Ma, I would also like to thank my lab mate Junan Zhu for numerous discussions about various research projects.

Finally, I want to express my special thanks to my parents and my fiance. They have always been by my side with financial and spiritual support throughout the entire journey of my doctoral studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Overview of Compressive Signal Reconstruction

Recent decades have been characterized by continuous radical developments in information technology, which have been deployed in many aspects of industrial applications and our daily lives. This has led to a deluge of data being acquired, processed, and stored, which has resulted in an increasing demand for hardware such as acquisition devices, data processors, and storage units. To reduce these hardware bottlenecks, it is desirable to compress the data as much as possible. At the same time, when the data is retrieved, we want to reconstruct it accurately.

To overcome these challenges, data compression is widely used. However, if the data can be compressed after acquisition, then it may also be possible for a reduced amount of data to be acquired, and this concept encourages the emergence of compressive sensing [7, 20, 31]. The intellectual foundations underlying compressive sensing rely on the ubiquitous compressibility of data: in an appropriate basis, most of the information contained in a set of data often resides in just a few large coefficients. The image compression standard JPEG (Joint Photographic Experts Group), for example, employs a discrete cosine transform (DCT) to approximate an image with a few large coefficients. It has been proved that, for a variety of settings, the information contained in the few large coefficients can be captured by a small number of random linear projections [44], and the data can then be reconstructed in a computationally feasible manner from these random projections [7, 20, 31].

Mathematically speaking, suppose that the data or signal of interest is organized as a vector $\mathbf{x}$ of length N, and the process of acquiring the signal $\mathbf{x}$ is approximately linear, which is modeled using a linear transform matrix $\mathbf{A}$ of dimension $M \times N$. Then we obtain the measurements $\mathbf{y}$ of dimension M as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z}, \tag{1.1}$$

where $\mathbf{z}$ is additive noise in the signal acquisition system. Our goal is to reconstruct or estimate the input signal $\mathbf{x}$ from the noisy linear measurements $\mathbf{y}$, using knowledge of the measurement matrix $\mathbf{A}$ and statistical information about the noise $\mathbf{z}$. Because the measurements $\mathbf{y}$ usually have a smaller dimension than $\mathbf{x}$, the signal is acquired in a compressive manner, and thus we call the process of reconstructing $\mathbf{x}$ *compressive signal reconstruction*.

In this dissertation, we explore compressive signal reconstruction methods, namely to reconstruct the input signal $\mathbf{x}$ from the noisy linear compressive measurements $\mathbf{y}$.

1.2 Outline

The remainder of the dissertation is arranged as follows.

In Chapter 2, we review two recent popular compressive signal reconstruction frameworks called generalized approximate message passing (GAMP) and approximate message passing (AMP). GAMP and AMP enjoy the flexibility of providing different reconstruction strategies, which inspires us to expand their capabilities in signal reconstruction.

In the last iterations of GAMP and AMP, some statistical information about the input signal is generated. Based on such statistical information, it is convenient to design various reconstruction algorithms, and simultaneously predict the reconstruction quality. On the other hand, although certain performance metrics are frequently used to assess the reconstruction quality either because of their effectiveness or their mathematical advantages, other user-defined performance metrics could also be evaluated based on specific practical needs. Therefore, we propose a signal reconstruction algorithm that is optimal in the sense of user-defined performance metrics. This metric optimal approach is described in Chapter 3.

Among the user-defined performance metrics, we find that minimizing the average ℓ_∞-norm error is closely related to the Wiener filter, a simple linear estimator in signal processing, which provides us interesting mathematical insight. In Chapter 4, we derive such mathematical properties in detail and embed the ℓ_∞-norm error minimization in GAMP and AMP.

Considering that compressive signal reconstruction has an important role in the imaging field, owing to its applications in seismic imaging, medical imaging, astronomy, and remote sensing, we further study the possibility of applying the AMP framework to compressive imaging in Chapter 5. AMP converts the compressive imaging problem to an image denoising problem at each iteration, and therefore, by designing an effective image denoiser that is suitable for AMP iterations, we come up with a compressive image reconstruction algorithm that is fast yet outperforms the prior art in terms of both reconstruction quality and runtime.

In Chapter 6, a practical compressive imaging problem, compressive hyperspectral imaging, is explored. The measurement matrix that models the hyperspectral imaging process is highly structured and not i.i.d. Gaussian. However, AMP suffers from a divergence problem when the measurement matrix goes beyond i.i.d. Gaussian. To overcome the divergence problem and maintain the reconstruction quality of the previously proposed AMP based algorithm, we modify the algorithm to fit the compressive hyperspectral imaging problem.

Finally, in Chapter 7, we conclude the dissertation and discuss possible directions for future work.

1.3 Notation

For the readers' convenience, a brief summary of notations that are used in this dissertation follows:

- $\mathbf{x}$: input signal of interest, and this is the signal to be reconstructed;
- $\mathbf{A}$: measurement matrix;
- $\mathbf{y}$: measurements;
- $\mathbf{z}$: additive noise in measurements;
- N: length of the input signal;
- M: number of measurements;
- $\mathbf{q}$: output of scalar channels;
- $\mathbf{v}$: noise in scalar channels produced by GAMP or AMP;
- t: iteration number;
- $\mathbf{r}$: residual in GAMP or AMP;
- $\mathbf{w}$: output of arbitrary scalar channels;
- $\mathbf{u}$: noise in arbitrary scalar channels;
- m, n, l: dimensions of hyperspectral image cubes;
- $(\cdot)^T$: the transpose of a vector or a matrix;
- $(\cdot)_i$: the i-th component of a vector;
- $\|\cdot\|_0$: the ℓ_0-norm of a vector, i.e., $\|\mathbf{x}\|_0$ is the number of nonzeros in $\mathbf{x}$;
- $\|\cdot\|_p$: the ℓ_p-norm of a vector (0¡p¡∞), i.e., $\|\mathbf{x}\|_p = (\sum_{i=1}^N |x_i|^p)^{\frac{1}{p}}$;

- $\| \cdot \|_\infty$: the ℓ_∞-norm of a vector, i.e., $\|\mathbf{x}\|_\infty = \max_{i=\{1,2,\ldots,N\}} |x_i|$;
- $\mathcal{N}(\mu, \sigma^2)$: Gaussian distribution with mean μ and variance σ^2.

Chapter 2

Review of Related Work

2.1 Basics of Compressive Signal Reconstruction

Let us first overview the mathematical justification of compressive signal reconstruction. Suppose that a signal $\mathbf{x}$ of dimension N is K-sparse, i.e., the vector $\mathbf{x}$ contains at most K nonzeros whereas the rest are zeros. Suppose further that the measurement matrix $\mathbf{A}$ satisfies the so-called restricted isometry property (RIP) [20]. Specifically, the RIP is a condition for $\mathbf{A}$ such that for some $\epsilon > 0$ and for any K-sparse vector $\mathbf{x}$,

$$(1-\epsilon) < \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} < (1+\epsilon). \tag{2.1}$$

In the noiseless case where $\mathbf{y} = \mathbf{A}\mathbf{x}$, there is no additive noise $\mathbf{z}$ as shown in equation (1.1), and it has been shown that if the matrix $\mathbf{A}$ satisfies the restricted isometry property (RIP) [20], then $\mathbf{x}$ can be exactly reconstructed from $\mathbf{y}$ with high probability by solving for

$$\widehat{\mathbf{x}} = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_0 \quad \textit{subject to} \quad \mathbf{A}\widetilde{\mathbf{x}} = \mathbf{y}. \tag{2.2}$$

Surprisingly, the above nonconvex optimization problem can be relaxed to the following convex optimization problem,

$$\widehat{\mathbf{x}} = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_1 \quad \textit{subject to} \quad \mathbf{A}\widetilde{\mathbf{x}} = \mathbf{y}. \tag{2.3}$$

In a more practical setting where additive noise $\mathbf{z}$ in the measurements exists, $\mathbf{x}$ can be reconstructed [106] by solving for

$$\widehat{\mathbf{x}} = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_1 \quad \textit{subject to} \quad \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 \leq \sigma^2, \tag{2.4}$$

where σ^2 is the noise variance. Typically, equation (2.4) is solved by introducing a Lagrange multiplier λ,

$$\widehat{\mathbf{x}} = \arg\min_{\widetilde{\mathbf{x}}} \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 + \lambda\|\widetilde{\mathbf{x}}\|_1. \tag{2.5}$$

There exists many algorithms that solve for equation (2.5), such as basis pursuit denoising (BPDN) [24], least absolute shrinkage and selection operator (LASSO) [108], and gradient projection for sparse reconstruction (GPSR) [38].

In fact, the solution $\widehat{\mathbf{x}}$ of equation (2.5) can be regarded as a maximum a-posteriori (MAP) estimator [60]. To see this, suppose that the additive noise $\mathbf{z}$ is independent and identically distributed (i.i.d.) Gaussian distributed, i.e.,

$$f(\mathbf{y}|\mathbf{A}\mathbf{x}) = \prod_{i=1}^{M} \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y_i - (\mathbf{A}\mathbf{x})_i)^2}{2\sigma^2}\right), \tag{2.6}$$

where $f(\cdot)$ denotes probability density function and $(\cdot)_i$ denotes the i-th component of a vector. Suppose further that the signal $\mathbf{x}$ is i.i.d. Laplacian distributed, i.e.,

$$f(\mathbf{x}) = \prod_{j=1}^{N} \frac{\lambda}{2}\exp\left(-\lambda x_j\right). \tag{2.7}$$

Under these assumptions, the MAP estimator $\widehat{\mathbf{x}}_{\text{MAP}}$ is

$$\begin{aligned}
\widehat{\mathbf{x}}_{\text{MAP}} &= \arg\max_{\mathbf{w}} f(\mathbf{x}|\mathbf{y}) && (2.8)\\
&= \arg\max_{\mathbf{w}} f(\mathbf{x}) \cdot f(\mathbf{y}|\mathbf{A}\mathbf{x})\\
&= \arg\max_{\mathbf{w}} \prod_{i=1}^{M} \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y_i - (\mathbf{A}\mathbf{x})_i)^2}{2\sigma^2}\right) \cdot \prod_{j=1}^{N} \frac{\lambda}{2}\exp\left(-\lambda x_j\right) && (2.9)\\
&= \arg\max_{\mathbf{w}} -\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 - \lambda\|\mathbf{x}\|_1 + \text{constant} && (2.10)\\
&= \arg\min_{\mathbf{w}} \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_1,
\end{aligned}$$

where (2.10) is derived by taking the logarithm of (2.9). Note that the assumption of $\mathbf{x}$ being Laplacian distributed is based on the assumption that $\mathbf{x}$ is sparse. It is also possible to assign other types of prior distributions to $\mathbf{x}$. When the prior distribution of $\mathbf{x}$ is not Laplacian, the MAP estimator (2.8) is no longer a solution of equation (2.5). Therefore, solving for (2.8) becomes difficult, because the variables in the two probability density functions $f(\mathbf{x})$ and $f(\mathbf{y}|\mathbf{A}\mathbf{x})$ have different lengths, and the statistical connection between the input signal $\mathbf{x}$ and the measurements $\mathbf{y}$ is mixed by the linear transform matrix $\mathbf{A}$. The generalized approximate message passing algorithm (GAMP) [78, 79] is an algorithmic framework that can efficiently approximate the posterior distribution $f(\mathbf{x}|\mathbf{y})$. Moreover, approximate message passing (AMP) is a line of work by Donoho et al. [33] that coincides with a special setting of GAMP. In Chapters 3 and 6, we will describe our proposed compressive signal reconstruction algorithms based on GAMP or AMP. Therefore, we overview GAMP and AMP below (Section 2.2).

One last remark before the overview of GAMP and AMP is that the signal $\mathbf{x}$ may not always be sparse explicitly, but sparse in an appropriate basis. Suppose that the signal $\mathbf{x}$ is sparse in a basis $\mathbf{\Psi}$, i.e., $\mathbf{x} = \mathbf{\Psi}\mathbf{s}$, where $\mathbf{\Psi} \in \mathbb{R}^{N\times N}$ and so $\mathbf{s}$ is sparse and of length N, then the measurements $\mathbf{y}$ become $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z} = \mathbf{A}\mathbf{\Psi}\mathbf{s} + \mathbf{z}$. Consequently, reconstructing $\mathbf{x}$ from $\mathbf{y}$ becomes reconstructing $\mathbf{s}$ from $\mathbf{y}$, and we can use equation (2.3) or equation (2.8) as mentioned above to solve for $\mathbf{s}$, but we require that $(\mathbf{A}\mathbf{\Psi})$ satisfies RIP.

2.2 Message Passing Algorithms

2.2.1 Generalized approximate message passing (GAMP)

Approximate message passing (AMP) can be regarded as a special case of generalized AMP (GAMP). We first take a look at the GAMP algorithm, and AMP will follow easily.

GAMP considers the following problem model. Let

$$\mathbf{h} = \mathbf{A}\mathbf{x}, \tag{2.11}$$

and the connection between measurements $\mathbf{y}$ and the vector $\mathbf{h}$ is characterized by a conditional distribution,

$$f_{\mathbf{Y}|\mathbf{H}}(\mathbf{y}|\mathbf{h}) = \prod_{i=1}^{M} f_{Y|H}(y_i|h_i). \tag{2.12}$$

Note that the system model in (2.11) and (2.12) is a generalized version of equation (1.1), because the connection between $\mathbf{y}$ and $\mathbf{A}\mathbf{x}$ is not necessarily additive in (2.11) and (2.12). GAMP is derived from belief Propagation (BP) [15], which is an iterative method used to compute the marginals of a Bayesian network, i.e., the marginals $f(x_j|\mathbf{y})$ for $j = \{1, 2, \ldots, N\}$. Consider the bipartite graph, called a *Tanner* or *factor* graph, shown in Figure 2.1, where circles represent random variables (called *variable nodes*), and related variables are connected through functions (represented by squares, called *factor nodes* or *function nodes*) that indicate dependencies. In standard BP, there are two types of messages passed through the nodes: messages from variable nodes to factor nodes, $m_{x\to y}$, and messages from factor nodes to variable nodes, $m_{y\to x}$. If we denote the set of function nodes connected to the variable x by $N(x)$, the set of variable nodes connected to the function y by $N(y)$, and the factor function at node y by Φ_y, then the two

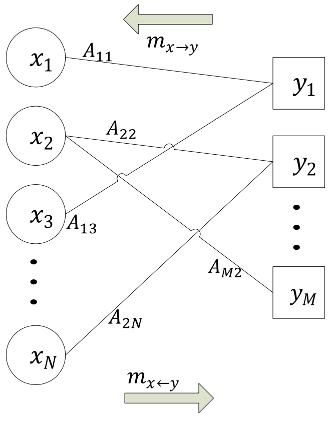


Figure 2.1: *Tanner graph for relaxed belief propagation.*

types of messages are defined as follows [15]:

$$\begin{aligned} m_{x\to y} &= \prod_{k\in N(x)\backslash y} m_{k\to x}, \\ m_{y\to x} &= \sum_{\ell\in N(y)\backslash x} \Phi_y m_{\ell\to y}. \end{aligned}$$

The basic BP idea described above can be applied to compressive sensing systems [76, 77]. In the Tanner graph, an input vector $x = (x_1, x_2, ..., x_N)^T$ is associated with the variable nodes (input nodes), and the output vector $y = (y_1, y_2, ..., y_M)^T$ is associated with the function nodes (output nodes). If $A_{ij} \neq 0$, then nodes x_j and y_i are connected to an edge (i, j), where the set of such edges E is defined as $E = \{(i, j) : A_{ij} \neq 0\}$.

In standard BP methods [9, 19, 50, 67], the distribution functions of x_j and h_i as well as the channel distribution function $f_{Y|H}(y_i|h_i)$ were set to be the messages passed along the graph, but it is difficult to compute those distributions, making the standard BP method computationally expensive. In [48], a simplified algorithm, called relaxed BP, was suggested. In this algorithm, means and variances replace the distribution functions themselves and serve as the messages passed through the nodes of the Tanner graph, greatly reducing the computational complexity. In [77, 78, 83], this method was extended to a more general case where the channel $f_{Y|H}$ is not necessarily Gaussian. The algorithm is called generalized approximate message passing (GAMP), and is summarized in Algorithm 1. In Algorithm 1, the estimate of $\mathbf{x}$ can be obtained according to either MAP or minimum mean square error (MMSE) criteria. In this dissertation, we focus on GAMP using an MMSE criterion, which is defined by functions $F_i(\cdot)$ and $G_j(\cdot)$ as follows,

$$
\begin{aligned}
F_i(p, \nu^p) \quad &\triangleq \quad \frac{\int h \exp(-f_i(h)) \mathcal{N}(h; p, \nu^p) dh}{\int \exp(-f_i(h)) \mathcal{N}(h; p, \nu^p) dh}, &(2.13)\\
G_j(q, \nu^q) \quad &\triangleq \quad \frac{\int x \exp(-g_j(x)) \mathcal{N}(x; q, \nu^q) dx}{\int \exp(-g_j(x)) \mathcal{N}(x; q, \nu^q) dx}. &(2.14)
\end{aligned}
$$

Algorithm 1 GAMP [17]

```
Inputs: $\forall i, j$: $F_i(\cdot)$, $G_j(\cdot)$, $x_j(1)$, $\nu_j^x(1)$ , $a_{ij}$, $T_{\max} \geq 1$, $\epsilon \geq 0$, $\beta_0 \in (0, 1]$
Outputs: $\forall j$: $x_j(t+1)$, $\nu_j^x(t+1)$
Initialization: $\forall i$: $s_i(0) = 0$, $t = 1$

  for $t = 1, \dots, T_{\max}$ do
     1. $\forall i : \nu_i^p(t) = \sum_{j=1}^N |a_{ij}|^2 \nu_j^x(t)$
     2. $\forall i : p_i(t) = \sum_{j=1}^N a_{ij} x_j(t) - \nu_i^p(t) r_i(t-1)$
     3. $\forall i : \nu_i^h(t) = \nu_i^p(t) F_i'(p_i(t), \nu_i^p(t))$
     4. $\forall i : h_i(t) = F_i(p_i(t), \nu_i^p(t))$
     5. $\forall i : \nu_i^r(t) = \left(1 - \frac{\nu_i^h(t)}{\nu_i^p(t)}\right) \frac{1}{\nu_i^p(t)}$
     6. $\forall i : r_i(t) = \frac{h_i(t) - p_i(t)}{\nu_i^p(t)}$
     7. $\forall j : \nu_j^q(t) = \frac{1}{\sum_{i=1}^M |a_{ij}|^2 \nu_i^r(t)}$
     8. $\forall j : q_j(t) = x_j(t) + \nu_j^q(t) \sum_{i=1}^M a_{ij}^* r_i(t)$
     9. $\forall j : \nu_j^x(t+1) = \nu_j^q(t) G_j'(q_j(t), \nu_j^q(t))$
    10. $\forall j : x_j(t+1) = G_j(q_j(t), \nu_j^q(t))$
  end for
```

At each iteration of Algorithm 1, the compressive sensing system in (2.11) and (2.12) is converted to scalar Gaussian channels in Line 8,

$$
q_j(t) = x_j(t) + v_j(t), \quad \forall j = 1, 2, \dots, N, \tag{2.15}
$$

where the Gaussian noise satisfies $v_j(t) \sim \mathcal{N}(0, \nu_j^q(t))$.

2.2.2 Approximate message passing (AMP)

A special yet frequently discussed setting in (1.1) is that the measurement noise is white Gaussian, which is equivalent to the conditional distribution in (2.12) being i.i.d. Gaussian as shown in (2.6). In this setting, GAMP reduces to approximate message passing (AMP) [33]. AMP proceeds iteratively according to

$$\mathbf{x}^{t+1} = \eta_t(\mathbf{A}^T\mathbf{r}^t + \mathbf{x}^t), \tag{2.16}$$

$$\mathbf{r}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \frac{1}{R}\mathbf{r}^{t-1}\langle \eta'_{t-1}(\mathbf{A}^T\mathbf{r}^{t-1} + \mathbf{x}^{t-1})\rangle, \tag{2.17}$$

where $R = M/N$ represents the measurement rate, $\eta_t(\cdot)$ is a denoising function at the t-th iteration, $\eta'_t(\mathbf{q}) = \frac{\partial}{\partial \mathbf{q}}\eta_t(\mathbf{q})$, and $\langle \mathbf{u} \rangle = \frac{1}{N}\sum_{i=1}^N u_i$ for some vector $\mathbf{u} = (u_1, u_2, \ldots, u_N)$. The last term in equation (2.17) is called the "Onsager reaction term" [33, 107] in statistical physics, and the corresponding Onsager reaction term in Algorithm 1 is the $\nu_i^p(t)r_i(t-1)$ term in Step 2.

Chapter 3

Signal Reconstruction with Arbitrary Error Metrics

We are now ready to introduce our proposed compressive signal reconstruction algorithms. In this chapter, we describe a reconstruction algorithm that achieves application-oriented signal reconstruction performance. This algorithm was proposed in Tan et al. [96, 99, 100]. Note that we referred to the algorithm by Rangan [77] as relaxed belief propagation in previous publications [96, 99, 100], whereas in this chapter we refer to the algorithm as "GAMP" for consistency throughout the dissertation.

In this chapter, we consider the system model defined in (2.11) and (2.12) where the input signal $\mathbf{x} \in \mathbb{R}^N$ is independent and identically distributed (i.i.d.), and the measurement matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is sparse and known (typically $M < N$). Our goal is to reconstruct $\mathbf{x}$ from $\mathbf{y}$ and $\mathbf{A}$. When we take some algorithm to reconstruct signals from their compressive measurements, the performance of the reconstruction algorithm is often characterized by some error metric that quantifies the distance between the estimated and the original signals. For a signal $\mathbf{x}$ and its estimate $\widehat{\mathbf{x}}$, both of length N, the error between them is the summation over the component-wise errors,

$$D(\widehat{\mathbf{x}}, \mathbf{x}) = \sum_{j=1}^{N} d(\widehat{x}_j, x_j). \tag{3.1}$$

For example, if the metric is absolute error, then $d(\widehat{x}_j, x_j) = |\widehat{x}_j - x_j|$; for squared error, $d(\widehat{x}_j, x_j) = (\widehat{x}_j - x_j)^2$.

Squared error is one of the most popular error metrics in various problems, due to many of its mathematical advantages. For example, minimum mean squared error (MMSE) estimation provides both variance and bias information about an estimator [45], and in the Gaussian case it is linear and thus often easy to implement [60]. However, there are applications where MMSE estimation is inappropriate. In medical image reconstruction, for example, it is important to obtain correct diagnoses in computed tomography scans and magnetic resonance imaging, and the rate of correct diagnoses may be an error metric of interest rather than the squared error. Moreover, MMSE estimation is sensitive to outliers [28, 119]. Therefore, alternative error metrics can be used instead.

In this chapter, we describe an algorithm that reconstructs signals by minimizing the average error metric of interest.

3.1 Related Work

As mentioned above, squared error is most commonly used as the error metric in estimation problems given by (2.6). Mean-squared optimal analysis and algorithms were introduced in [9, 46–48, 81] to estimate a signal from measurements corrupted by Gaussian noise; in [49, 77, 78], further discussions were made about the circumstances where the output channel is arbitrary, while, again, the MMSE estimator was put forth. Another line of work, based on a greedy algorithm called *orthogonal matching pursuit*, was presented in [71, 109] where the mean squared error decreases over iterations. Absolute error is also under intense study in signal estimation. For example, an efficient sparse recovery scheme that minimizes the absolute error was provided in [11, 55]; in [27], a fundamental analysis was offered on the minimal number of measurements required while keeping the estimation error within a certain range, and absolute error was one of the metrics concerned. Support recovery error is another metric of great importance, for example because it relates to properties of the measurement matrices [118]. The authors of [110, 117, 118] discussed the support error rate when recovering a sparse signal from its noisy measurements; support-related performance metrics were applied in the derivations of theoretical limits on the sampling rate for signal recovery [1, 85]. The readers may notice that previous work only paid attention to limited types of error metrics. What if absolute error, cubic error, or other non-standard metrics are required in a certain application?

Therefore, (*i*) we suggest a pointwise Bayesian estimation algorithm that minimizes an arbitrary additive error metric; (*ii*) we prove that the algorithm is optimal; (*iii*) we study the fundamental information-theoretic performance limit of estimation for a given metric; (*iv*) we derive the performance limits for minimum mean absolute error, minimum mean support error, and minimum mean weighted-support error estimators, and obtain the receiver operating characteristic (ROC) of the modeled system by weighted-support error. This algorithm is based on the assumption that GAMP [78] converges to a set of degraded scalar Gaussian channels [46, 47, 49, 81]. GAMP with the MMSE criterion is well-known to be optimal for the squared error, while we further show that GAMP can do more – because the sufficient statistics are given, other additive error metrics can also be minimized with one more simple and fast step. This is convenient for users who desire to recover the original signal with a non-standard additive error metric. Simulation results show that our algorithm outperforms algorithms such as GAMP [77], which is optimal for squared error, and compressive sampling matching pursuit (CoSaMP) [71], a greedy reconstruction algorithm. Moreover, we compare our algorithm with the suggested theoretical limits for minimum mean absolute error (MMAE), minimum mean support error (MMSuE), and minimum mean weighted-support error (MMWSE), and illustrate that our algorithm is optimal.

3.2 Estimation Algorithm

Recall that in Chapter 2 the GAMP algorithm generates two sequences, $q_j(t)$ and $\nu_j^q(t)$, in equation (2.15), where $t \in \mathbb{Z}^+$ denotes the iteration number. Under the assumptions that the signal dimension $N \to \infty$, the iteration number $t \to \infty$, and the ratio M/N is fixed, the sequences $q_j(t)$ and $\nu_j^q(t)$ converge to sufficient statistics for the compressive measurements $\mathbf{y}$ (2.12). More specifically, in the large system limit, the conditional distribution $f(x_j|q_j(t), \nu_j^q(t))$ converges to the conditional distribution $f(x_j|\mathbf{y})$, where $q_j(t)$ can be regarded as a Gaussian-noise-corrupted version of x_j, and $\nu_j^q(t)$ is the noise variance. It has been shown [77] that $\nu_j^q(t))$ converges to a fixed point that satisfies Tanaka's equation, which has been analyzed in detail (cf. [33, 46, 47, 67, 78, 81, 105]). We define the limits of the two sequences,

$$\begin{aligned} \lim_{t\to\infty} q_j(t) &= q_j, \\ \lim_{t\to\infty} \nu_j^q(t) &= \nu, \end{aligned}$$

for $j = 1, 2, \dots, N$. We now simplify equation (2.15) as follows,

$$q_j = x_j + v_j, \tag{3.2}$$

where $v_j \sim \mathcal{N}(0, \nu)$ for $j = 1, 2, \dots, N$.

The structure of our metric-optimal algorithm is illustrated in the dashed box in Figure 3.1. The inputs of the algorithm are: (*i*) a distribution function $f_{\mathbf{X}}(\mathbf{x})$, the prior of the original input $\mathbf{x}$; (*ii)* a vector $\mathbf{q} = (q_1, q_2, ..., q_N)$, the outputs of the scalar Gaussian channels computed by GAMP [78]; (*iii*) a scalar ν, the variance of the Gaussian noise in (3.2); and (*iv*) an error metric function $D(\widehat{\mathbf{x}}, \mathbf{x})$ specified by the user. The vector $\mathbf{q}$ and the scalar ν are the outputs of GAMP [78], and in particular we generate $\mathbf{q}$ and ν using the software package "GAMP" [80].

Because the scalar channels have additive Gaussian noise, and the variances of the noise are

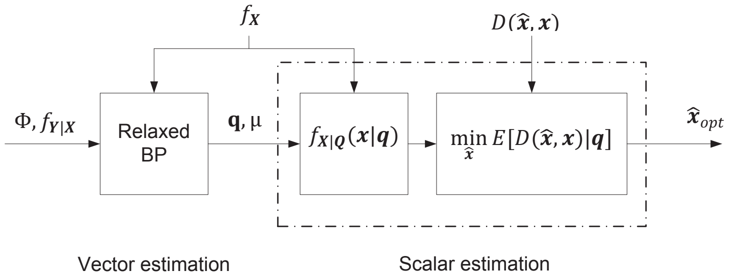


Figure 3.1: *The structure of the metric-optimal estimation algorithm.*

all ν, we can compute the conditional probability density function $f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q})$ from Bayes' rule:

$$\begin{aligned} f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q}) &= \frac{f_{\mathbf{Q}|\mathbf{X}}(\mathbf{q}|\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{f_{\mathbf{Q}}(\mathbf{q})} \\ &= \frac{f_{\mathbf{Q}|\mathbf{X}}(\mathbf{q}|\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{\int f_{\mathbf{Q}|\mathbf{X}}(\mathbf{q}|\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}}, \end{aligned} \tag{3.3}$$

where

$$f_{\mathbf{Q}|\mathbf{X}}(\mathbf{q}|\mathbf{x}) = \frac{1}{\sqrt{(2\pi\nu)^N}} \exp\left(-\frac{\|\mathbf{q}-\mathbf{x}\|_2^2}{2\nu}\right).$$

Given an error metric $D(\widehat{\mathbf{x}}, \mathbf{x})$, the optimal estimand $\widehat{\mathbf{x}}_{\text{opt}}$ is generated by minimizing the conditional expectation of the error metric $E[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}]$, which is easy to compute using $f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q})$:

$$E[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}] = \int D(\widehat{\mathbf{x}}, \mathbf{x}) f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q}) d\mathbf{x}.$$

Then,

$$\begin{aligned} \widehat{\mathbf{x}}_{\text{opt}} &= \arg\min_{\widehat{\mathbf{x}}} E[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}] \\ &= \arg\min_{\widehat{\mathbf{x}}} \int D(\widehat{\mathbf{x}}, \mathbf{x}) f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q}) d\mathbf{x}. \end{aligned} \tag{3.4}$$

The conditional probability $f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q})$ is separable, because the parallel scalar Gaussian channels (3.2) are separable and $f_{\mathbf{X}}(\mathbf{x})$ is i.i.d. Moreover, the error metric function $D(\widehat{\mathbf{x}}, \mathbf{x})$ (3.1) is also separable. Therefore, the problem reduces to scalar estimation [60],

$$\begin{aligned} \widehat{x}_{\text{opt},j} &= \arg\min_{\widehat{x}_j} E[d(\widehat{x}_j, x_j)|q_j] \\ &= \arg\min_{\widehat{x}_j} \int d(\widehat{x}_j, x_j) f_{x_j|q_j}(x_j|q_j) dx_j, \end{aligned} \tag{3.5}$$

for $j = 1, 2, \ldots, N$. Equation (4.14) minimizes a single-variable function. In Section 3.4, we show how to perform this minimization in three example cases.

3.3 Theoretical Results

Having discussed the algorithm, we now provide a theoretical justification for its performance.

Claim 1 *Given the system model described by* (2.11), (2.12) *and an error metric* $D(\widehat{\mathbf{x}}, \mathbf{x})$ *of the form defined by* (3.1)*, as the signal dimension* $N \to \infty$ *and the measurement ratio* M/N *is fixed, the optimal estimand of the input signal is given by*

$$\widehat{\mathbf{x}}_{opt} = \arg\min_{\widehat{\mathbf{x}}} E\left[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}\right],$$

where the vector entries $\mathbf{q} = (q_1, q_2, \ldots, q_N)$ *are the outputs of the scalar Gaussian channels* (3.2)*.*

The rationale for Claim 1 is as follows. Because the probability density function $f_{X_j|\mathbf{Y}}(x_j|\mathbf{y})$ is statistically equivalent to $f_{X_j|Q_j}(x_j|q_j)$ in the large system limit, once we know the value of ν, estimating each x_j from all channel outputs $\mathbf{y} = (y_1, y_2, ..., y_M)$ is equivalent to estimating x_j from the corresponding scalar channel output q_j. GAMP [78] calculates the sufficient statistics q_j and ν. Therefore, an estimator based on minimizing the conditional expectation of the error metric, $E\left[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}\right]$, provides an asymptotically optimal result.

Claim 1 states that, in the large system limit, the estimator satisfying (4.13) is optimal, because it minimizes the conditional expectation of the error metric. The key point in the estimation problem is to obtain the posterior $f_{X_j|\mathbf{Y}}$. Fortunately, GAMP provides an asymptotically optimal method to decouple the mixing channels and thus an equivalent posterior $f_{X_j|Q_j}(x_j|q_j)$ can be computed easily, and our algorithm utilizes this convenient feature.

Following Claim 1, we can compute the minimum expected error achievable by *any* estimation algorithm for any additive error metric $D(\widehat{\mathbf{x}}_{\text{opt}}, \mathbf{x})$. This minimum expected error is the fundamental information-theoretic performance limit of interest in this problem; no estimation algorithm can out-perform this limit. At the same time, we will see in Section 3.4 that, for three example error metrics, our BP-based algorithm *matches* the performance of the information-theoretic limit, and is thus optimal.

Claim 2 *For a system modeled by* (2.11), (2.12)*, as the signal dimension* $N \to \infty$*, the minimum mean user-defined error (MMUE) is given by*

$$MMUE(f_{\mathbf{X}}, \nu) = \int_{R(\mathbf{Q})} \left(\int_{R(\mathbf{X})} D(\widehat{\mathbf{x}}_{opt}, \mathbf{x}) \left(\frac{1}{\sqrt{(2\pi\nu)^N}} exp\left(-\frac{\|\mathbf{q} - \mathbf{x}\|^2}{2\nu} \right) \right) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \right) d\mathbf{q}, \quad (3.6)$$

where the optimal estimand $\widehat{\mathbf{x}}_{opt}$ is determined by (4.13)*, $R(\cdot)$ represents the range of a variable, and ν is the variance of the noise of the scalar Gaussian channel* (3.2)*.*

Equation (3.6) can be derived in the following steps.

$$\begin{aligned}
& \mathrm{MMUE}(f_{\mathbf{X}},\nu) = E[D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x})] \\
= & \int_{R(\mathbf{Q})} E_{\mathbf{Q}}\Big[E[D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x})|\mathbf{q}]\Big] f_{\mathbf{Q}}(\mathbf{q})d\mathbf{q} \\
= & \int_{R(\mathbf{Q})} E[D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x})|\mathbf{q}] f_{\mathbf{Q}}(\mathbf{q})d\mathbf{q} \\
= & \int_{R(\mathbf{Q})} \left(\int_{R(\mathbf{X})} D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x}) f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q})d\mathbf{x}\right) f_{\mathbf{Q}}(\mathbf{q})d\mathbf{q} \\
= & \int_{R(\mathbf{Q})} \left(\int_{R(\mathbf{X})} D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x}) \frac{f_{\mathbf{Q}|\mathbf{X}}(\mathbf{q}|\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{f_{\mathbf{Q}}(\mathbf{q})} d\mathbf{x}\right) f_{\mathbf{Q}}(\mathbf{q})d\mathbf{q} \\
= & \int_{R(\mathbf{Q})} \left(\int_{R(\mathbf{X})} D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x}) f_{\mathbf{Q}|\mathbf{X}}(\mathbf{q}|\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})d\mathbf{x}\right) d\mathbf{q} \\
= & \int_{R(\mathbf{Q})} (\int_{R(\mathbf{X})} D(\widehat{\mathbf{x}}_{\mathrm{opt}},\mathbf{x}) \frac{1}{\sqrt{(2\pi\nu)^N}} \exp\left(-\frac{\|\mathbf{q}-\mathbf{x}\|^2}{2\nu}\right) f_{\mathbf{X}}(\mathbf{x})d\mathbf{x})d\mathbf{q}.
\end{aligned}$$

Using both claims, we further analyze the estimation performance limits for three example error metrics in Sections 3.4.

3.4 Examples

3.4.1 Absolute error

Because the MMSE is the mean of the conditional distribution, the outliers in the set of data may corrupt the estimation, and in this case the minimum mean absolute error (MMAE) is a good alternative. For absolute error, $d_{\mathrm{AE}}(\widehat{x_j}, x_j) = |\widehat{x_j} - x_j|$, and we have the following corollary describing the performance limit of an MMAE estimator, where the proof is given in Appendix A.

Corollary 1 *For a system modeled by* (2.11), (2.12)*, as the signal dimension $N \to \infty$, the minimum mean absolute error (MMAE) estimator achieves*

$$MMAE(f_{\mathbf{X}},\nu) = N\int_{-\infty}^{+\infty}\left(\int_{-\infty}^{\widehat{x}_{j,MMAE}}(-x_j)f_{X_j|Q_j}(x_j|q_j)dx_j + \int_{\widehat{x}_{j,MMAE}}^{+\infty} x_j f_{X_j|Q_j}(x_j|q_j)dx_j\right) f_{Q_j}(q_j)dq_j, \tag{3.7}$$

where x_j (respectively, q_j) is the input (respectively, output) of the scalar Gaussian channel (3.2)*, $\widehat{x}_{j,MMAE}$ satisfies $\int_{\widehat{x}_{j,MMAE}}^{+\infty} f_{X_j|Q_j}(x_j|q_j)dx_j = \frac{1}{2}$, and $f_{X_j|Q_j}(x_j|q_j)$ is a function of f_{X_j} following* (3.3)*.*

3.4.2 Support recovery error

In some applications in compressive sensing, correctly estimating the locations where the data has nonzero values is almost as important as estimating the exact values of the data; it is a standard model selection error criterion [118]. The process of estimating the nonzero locations is called *support recovery.* Support recovery error is defined as follows, and this metric function is discrete,

$$d_{\text{support}}(\widehat{x}_j, x_j) = \text{xor}(\widehat{x}_j, x_j),$$

where

$$\text{xor}(\widehat{x}_j, x_j) = \begin{cases} 0, \text{ if } x_j = 0 \text{ and } \widehat{x}_j = 0 \\ 0, \text{ if } x_j \neq 0 \text{ and } \widehat{x}_j \neq 0 \\ 1, \text{ if } x_j = 0 \text{ and } \widehat{x}_j \neq 0 \\ 1, \text{ if } x_j \neq 0 \text{ and } \widehat{x}_j = 0 \end{cases}.$$

Corollary 2 *For a system modeled by* (2.11)*,* (2.12)*, where $f_{\mathbf{X}}$ is an i.i.d. sparse Gaussian prior such that* $\Pr(X_j \neq 0) = p$ *and $X_j \neq 0 \sim \mathcal{N}(0, \sigma^2)$, as the signal dimension $N \to \infty$, the minimum mean support error (MMSuE) estimator achieves*

$$MMSuE(f_{\mathbf{X}}, \nu) = N \cdot (1-p) \cdot erfc\left(\sqrt{\frac{\tau}{2\nu}}\right) + N \cdot p \cdot erf\left(\sqrt{\frac{\tau}{2(\sigma^2+\nu)}}\right), \tag{3.8}$$

where erf$(\cdot)$ and erfc$(\cdot)$ are the error function and complementary error function, respectively, and

$$\tau = 2 \cdot \frac{\sigma^2 + \nu}{\sigma^2/\nu} \cdot \ln\left(\frac{(1-p)\sqrt{\sigma^2/\nu + 1}}{p}\right).$$

Corollary 2 is proved in Appendix B.

3.4.3 Weighted-support error

In Section 3.4.2, we put equal weights on the error patterns *(i)* $\widehat{x}_j \neq 0$ while $x_j = 0$; and *(ii)* $\widehat{x}_j = 0$ while $x_j \neq 0$. In this section, we further put unequal weights on these two error patterns, because in many applications one of the error patterns is more important than the other. For example, suppose that $x_j \neq 0$ means a patient has some disease and $\widehat{x}_j = 0$ means the diagnosis shows no disease, then the error pattern $\widehat{x}_j = 0$ while $x_j \neq 0$ is more important than the other error pattern and needs to be avoided. Therefore, more weight can be put on the error pattern $\widehat{x}_j = 0$ while $x_j \neq 0$. We first define *false positive* error d_{FP} as

$$d_{\mathrm{FP}}(\widehat{x}_j, x_j) = d(\widehat{x}_j = 1, x_j = 0) = 1,$$

and *false negative* error d_{FN} as

$$d_{\mathrm{FN}}(\widehat{x}_j, x_j) = d(\widehat{x}_j = 0, x_j = 1) = 1.$$

Else the patterns coincide,

$$d(\widehat{x}_j = 0, x_j = 0) = d(\widehat{x}_j = 1, x_j = 1) = 0.$$

We then define *weighted-support error* as

$$d_{\mathrm{w\_support}}(\widehat{x}_j, x_j) = \beta \cdot d_{\mathrm{FP}}(\widehat{x}_j, x_j) + (1-\beta) \cdot d_{\mathrm{FN}}(\widehat{x}_j, x_j), \tag{3.9}$$

where $0 \leq \beta \leq 1$, and we set $d_{\mathrm{w\_support}}(\widehat{x}_j, x_j)$ as the error metric that we want to minimize. The false positive rate (or *false alarm rate*) is defined as $\Pr(\widehat{x}_j \neq 0 | x_j = 0)$, and the false negative rate (or *misdetection rate*) is defined as $\Pr(\widehat{x}_j = 0 | x_j \neq 0)$.

Corollary 3 *For a system modeled by (2.11), (2.12), where $f_{\mathbf{X}}$ is an i.i.d. sparse Gaussian prior such that $\Pr(X_j \neq 0) = p$ and $X_j \neq 0 \sim \mathcal{N}(0, \sigma^2)$, as the signal dimension $N \to \infty$,*

1. *The minimum mean weighted-support error (MMWSE) estimator achieves*

$$\mathit{MMWSE}(f_{\mathbf{X}}, \nu) = N\beta(1-p) \cdot \mathit{erfc}\left(\sqrt{\frac{\tau'}{2\nu}}\right) + N(1-\beta)p \cdot \mathit{erf}\left(\sqrt{\frac{\tau'}{2(\sigma^2+\nu)}}\right), \tag{3.10}$$

where

$$\tau' = 2 \cdot \frac{\sigma^2 + \nu}{\sigma^2/\nu} \cdot \ln\left(\frac{\beta(1-p)\sqrt{\sigma^2/\nu + 1}}{(1-\beta)p}\right).$$

2. *The false positive rate is*

$$\Pr(\widehat{x}_j \neq 0 | x_j = 0) = \mathit{erfc}\left(\sqrt{\frac{\tau'}{2\nu}}\right), \tag{3.11}$$

and the false negative rate is

$$\Pr(\widehat{x}_j = 0 | x_j \neq 0) = erf\left(\sqrt{\frac{\tau'}{2(\sigma^2 + \nu)}}\right). \tag{3.12}$$

The proof of Corollary 3 is provided in Appendix C.

It is shown in Corollary 3 that $\Pr(\widehat{x}_j \neq 0 | x_j = 0)$ and $\Pr(\widehat{x}_j = 0 | x_j \neq 0)$ vary when the value of β varies. Moreover, Comparing equation (3.10) to (3.8) in Corollary 2, the only difference is that τ is replaced by τ', which is the *decision threshold* that determines whether an estimand is zero or nonzero. That said, putting different weights on false positive error and false negative error is analogous to tuning the decision threshold, and thus trading off between the false alarm rate and the misdetection rate [60]. A *receiver operating characteristic* (ROC) curve [60] is shown in Section 6.4.

3.5 Numerical Results

Some numerical results are shown in this section to illustrate the performance of our estimation algorithm when minimizing a user-defined error metric. The Matlab implementation of our algorithm can be found at `http://people.engr.ncsu.edu/dzbaron/software/arb_metric/`.

We test our estimation algorithm on two linear systems modeled by (2.11) and (2.12): (*i*) Gaussian input and Gaussian channel; (*ii*) Weibull input and Poisson channel. In both cases, the input's length N is 10,000, and its sparsity rate is 3%, meaning that the entries of the input vector are nonzero with probability 3%, and zero otherwise. The matrix $\mathbf{A}$ we use is Bernoulli(0.5) distributed, and is normalized to have unit-norm rows. In the first case, the nonzero input entries are $\mathcal{N}(0,1)$ distributed, and the Gaussian noise is $\mathcal{N}(0, 3 \cdot 10^{-4})$ distributed, i.e., the signal to noise ratio (SNR) is 20 dB. In the second case, the nonzero input entries are Weibull distributed,

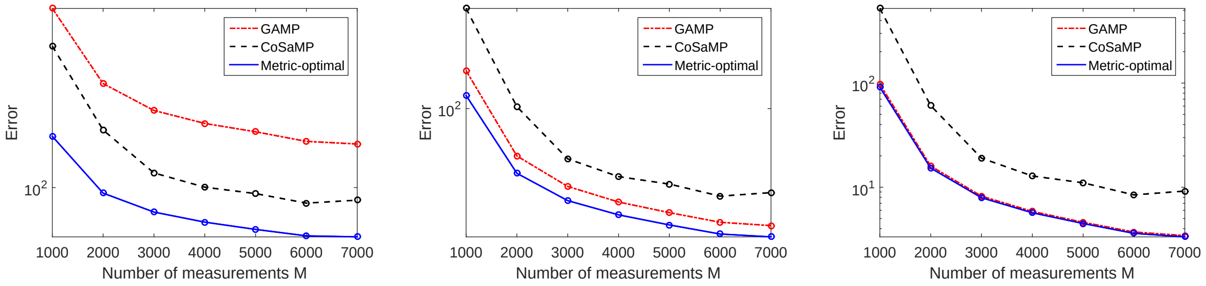
$$f(x_j; \lambda, k) = \begin{cases} \frac{k}{\lambda}\left(\frac{x_j}{\lambda}\right)^{k-1} e^{-(x_j/\lambda)^k} & x_j \geq 0 \\ 0 & x_j < 0 \end{cases},$$

where $\lambda = 1$ and $k = 0.5$. The Poisson channel is

$$f_{Y|W}(y_i|w_i) = \frac{(\alpha w_i)^{y_i} e^{-(\alpha w_i)}}{y_i!}, \quad \text{for all } i \in \{1, 2, \ldots, M\},$$

where the scaling factor of the input is $\alpha = 100$.

In order to illustrate that our estimation algorithm is suitable for reasonable error metrics,



(a) (b) (c)

Figure 3.2: *Comparison of the metric-optimal estimation algorithm, GAMP, and CoSaMP. Sparse Gaussian input and Gaussian channel; sparsity rate = 3%; input length $N = 10,000$; SNR = 20 dB. (a) $D(\widehat{\mathbf{x}}, \mathbf{x}) = \sum_{j=1}^{N} |\widehat{x}_j - x_j|^{0.5}$; (b) $D(\widehat{\mathbf{x}}, \mathbf{x}) = \sum_{j=1}^{N} |\widehat{x}_j - x_j|$; and (c) $D(\widehat{\mathbf{x}}, \mathbf{x}) = \sum_{j=1}^{N} |\widehat{x}_j - x_j|^{1.5}$.*

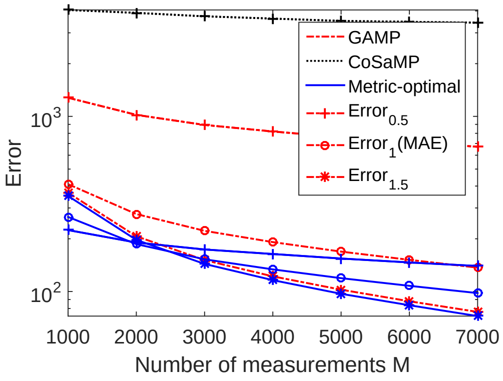


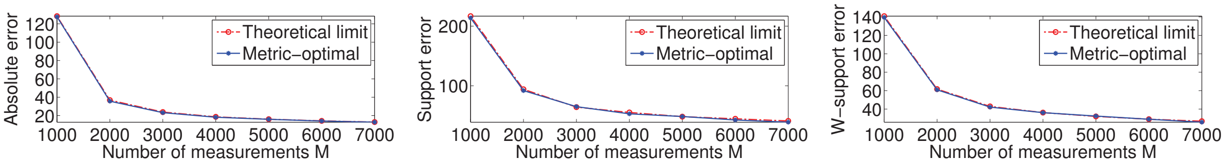
Figure 3.3: *Comparison of the metric-optimal estimation algorithm, GAMP, and CoSaMP. The "MAE" and the "Error$_{1.5}$" lines for "CoSaMP" appear beyond the scope of the vertical axis. Sparse Weibull input and Poisson channel; sparsity rate = 3%; input length $N = 10,000$; input scaling factor $\alpha = 100$.*

we considered absolute error and two other non-standard metrics:

$$\text{Error}_p = \sum_{j=1}^{N} |\widehat{x}_j - x_j|^p,$$

where $p = 0.5$ or 1.5.

We compare our algorithm with GAMP [78] and CoSaMP [71] algorithms. In Figure 3.2 and Figure 3.3, lines marked with "metric-optimal" present the errors of our estimation algorithm, and lines marked with "GAMP" (respectively, "CoSaMP") show the errors of the GAMP (respectively, CoSaMP) algorithm. Each point in the figure is an average of 100 experiments with the same parameters. Because the Poisson channel is not an additive noise channel and



(a) (b) (c)

Figure 3.4: *Comparisons of the metric-optimal estimators and the corresponding theoretical limits* (3.7), (3.8), *and* (3.10). *The corresponding two lines are on top of each other. Sparse Gaussian input and Gaussian channel; sparsity rate* = 3%; *input length* $N = 10{,}000$; *SNR* = 20 *dB. (a) Absolute error; (b) Support error; and (c) Weighted-support error.*

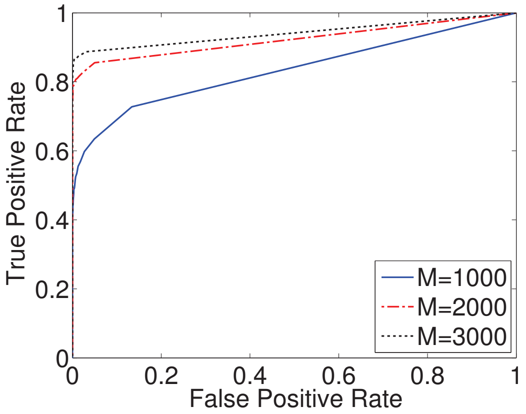


Figure 3.5: *The ROC curve obtained by setting weighted-support error as the error metric. Sparse Gaussian input and Gaussian channel; sparsity rate* = 3%; *input length* $N = 10{,}000$; *SNR* = 20 *dB.*

is not suitable for CoSaMP, the "MAE" and the "Error$_{1.5}$" lines for "CoSaMP" in Figure 3.3 appear beyond the scope of the vertical axis. It can be seen that our metric-optimal algorithm outperforms the other two methods.

To demonstrate the theoretical analysis of our algorithm in Sections 3.4, we compare our MMAE estimation results with the theoretical limit (3.7) in Figure 3.4a, where the integrations are computed numerically. In Figure 3.4b, we compare our MMSuE estimator with the theoretical limit (3.8), where the value of ν is acquired numerically from GAMP [80] with 20 iterations. In Figure 3.4c, our MMWSE estimator and its theoretical limit (3.10) are compared, where we fix the weight $\beta = 0.3$, and obtain the value of ν as in Figure 3.4b. In all three figures, each point on the "metric-optimal" line is generated by averaging 40 experiments with the same parameters. It is shown in all figures that the two lines are on top of each other. Therefore our estimation algorithm reaches the corresponding theoretical limits and is optimal.

Figure 3.5 illustrates the ROC curve obtained by setting the weighted-support error (3.9) as the error metric. We vary the value of β in (3.9) from 0 to 1, and compute the false positive rate as well as the false negative rate from (3.11) and (3.12). The ROC curve is a $\Pr(\widehat{x}_j \neq 0 | x_j = 0)$ (3.11) versus $\Pr(\widehat{x}_j \neq 0 | x_j \neq 0)$ plot, where $\Pr(\widehat{x}_j \neq 0 | x_j \neq 0)$ is called the *true positive rate*, and $\Pr(\widehat{x}_j \neq 0 | x_j \neq 0) = 1 - \Pr(\widehat{x}_j = 0 | x_j \neq 0)$ (3.12). In order to obtain different curves, we tune the number of measurements M, while keeping the sparsity rate, input length, and the SNR fixed. It can be seen that for the same level of false positive rate, a greater number of measurements achieves a higher true positive rate.

3.6 Conclusion

In this chapter, we introduced a pointwise estimation algorithm that deals with arbitrary additive error metrics in noisy compressive sensing. We verified that the algorithm is optimal in the large system limit, and provided a general method to compute the minimum expected error achievable by any estimation algorithm for a user-defined additive error metric. We started with the scalar Gaussian channel model of GAMP and extended it to a method that is applicable to any user-defined additive error metric. We discussed three error metric examples, absolute error, support error, and weighted-support error, and gave the theoretical performance limits for them. We further obtained the ROC curve for the modeled system by minimizing the weighted-support error. We also illustrated numerically that our algorithm reaches the three example theoretical limits, and outperforms GAMP and CoSaMP methods.

Chapter 4

Signal Reconstruction with Infinity Norm Error

Continuing our topic in Chapter 3 on signal reconstruction with arbitrary error metrics, we focus on a specific error metric in this chapter, ℓ_∞-norm error, also called worst-case error or infinity norm error. The results in this chapter appeared in Tan et al. [97, 98]. In contrast to the ℓ_2-norm error, which ensures that the reconstructed signal has low square error on average, the ℓ_∞-norm error ensures that every reconstructed signal component is close to the corresponding original signal component. In problems such as image and video compression [30] where the reconstruction quality at every signal component is important, it might be better to optimize for ℓ_∞-norm error. Our interest in the ℓ_∞-norm error is also motivated by applications including wireless communications [94], group testing [42], and trajectory planning in control systems [34], where we want to decrease the worst-case sensitivity to noise.

4.1 Gaussian Mixture Source

The Gaussian distribution is widely used to describe the probability densities of various types of data, owing to its advantageous mathematical properties [73]. It has been shown that non-Gaussian distributions can often be sufficiently approximated by an infinite mixture of Gaussians [2], so that the mathematical advantages of the Gaussian distribution can be leveraged when discussing non-Gaussian signals [2, 13, 95, 112, 113].

The main results in this chapter are based on the assumption that the input signal $\mathbf{x}$ is generated by an independent and identically distributed (i.i.d.) Gaussian mixture source,

$$x_i \sim \sum_{k=1}^{K} s_k \cdot \mathcal{N}(\mu_k, \sigma_k^2) = \sum_{k=1}^{K} \frac{s_k}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(x_i-\mu_k)^2}{2\sigma_k^2}}, \tag{4.1}$$

where the subscript $(\cdot)_i$ denotes the i-th component of a sequence (or a vector), $\mu_1, \mu_2, \ldots, \mu_K$ (respectively, σ_1^2, σ_2^2, $\ldots$,σ_K^2) are the means (respectively, variances) of the Gaussian components, and $0 < s_1, s_2, \ldots, s_K < 1$ are the probabilities of the K Gaussian components. Note that $\sum_{k=1}^{K} s_k = 1$. A special case of the Gaussian mixture is Bernoulli-Gaussian,

$$x_i \sim s \cdot \mathcal{N}(\mu_x, \sigma_x^2) + (1-s) \cdot \delta(x_i), \tag{4.2}$$

for some $0 < s < 1$, μ_x, and σ_x^2, where $\delta(\cdot)$ is the delta function [73]. The zero-mean Bernoulli-Gaussian model is often used in sparse signal processing [20, 31, 49, 50, 76, 77, 93, 111].

4.2 Problem Setting

We start our analysis on minimizing ℓ_∞-norm error for parallel Gaussian channels, and the results can be extended to compressive sensing models (1.1). In parallel Gaussian channels [13, 95], we consider

$$\mathbf{w} = \mathbf{x} + \mathbf{u}, \tag{4.3}$$

where $\mathbf{w}, \mathbf{x}, \mathbf{u} \in \mathbb{R}^N$ are the output signal, the input signal, and the additive white Gaussian noise (AWGN), respectively. The AWGN channel can be described by the conditional distribution

$$f_{\mathbf{W}|\mathbf{X}}(\mathbf{w}|\mathbf{x}) = \prod_{i=1}^{N} f_{W|X}(w_i|x_i) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma_u^2}} \exp\left(-\frac{(w_i - x_i)^2}{2\sigma_u^2}\right), \tag{4.4}$$

where σ_u^2 is the variance of the Gaussian noise. Note that we use different notations in (4.4) from the notations in (2.15) to emphasize that the parallel Gaussian channels we consider in this chapter do not necessarily come from GAMP or AMP.

Our goal is to reconstruct or estimate the original input signal $\mathbf{x}$ from the parallel Gaussian channel outputs $\mathbf{w}$ in (4.3). To evaluate how accurate the reconstruction process is, we quantify the ℓ_∞-norm error between $\mathbf{x}$ and its estimate $\widehat{\mathbf{x}}$,

$$\|\widehat{\mathbf{x}} - \mathbf{x}\|_\infty = \max_{i \in \{1,\ldots,N\}} |\widehat{x}_i - x_i|;$$

this error metric helps prevent any significant errors during the reconstruction process. The estimator that minimizes the expected value of $\|\widehat{\mathbf{x}} - \mathbf{x}\|_\infty$ is called the minimum mean ℓ_∞-norm error estimator. We denote this estimator by $\widehat{\mathbf{x}}_{\ell_\infty}$, which can be expressed as

$$\widehat{\mathbf{x}}_{\ell_\infty} = \arg\min_{\widehat{\mathbf{x}}} E\left[\|\widehat{\mathbf{x}} - \mathbf{x}\|_\infty\right]. \tag{4.5}$$

4.3 Related Work

Gaussian mixtures are widely used to model various types of signals, and a number of signal reconstruction methods have been introduced to take advantage of the Gaussian mixture distribution. For example, an infinite Gaussian mixture model was proposed in [2] to represent real data such as images, and a denoising scheme based on local linear estimators was developed to reconstruct the original data. A similar algorithm based on an adaptive Wiener filter was applied to denoise X-ray CT images [95], where a Gaussian mixture model was utilized. However, these works only quantified the ℓ_2-norm error of the denoising process. Signal reconstruction problems with ℓ_∞-norm error have not been well-explored, but there have been studies on general properties of the ℓ_∞-norm. For example, in Clark [26], the author developed a deductive method to calculate the distribution of the greatest element in a finite set of random variables; and Indyk [54] discussed how to find the nearest neighbor of a point while taking the ℓ_∞-norm distance into consideration.

4.4 Main Results

For parallel Gaussian channels (4.3), the minimum mean squared error estimator, denoted by $\widehat{\mathbf{x}}_{\ell_2}$, is achieved by the conditional expectation $E[\mathbf{x}|\mathbf{w}]$. If the input signal $\mathbf{x}$ is i.i.d. Gaussian (not a Gaussian mixture), i.e., $x_i \sim \mathcal{N}(\mu_x, \sigma_x^2)$, then the estimate

$$\widehat{\mathbf{x}}_{\ell_2} = E[\mathbf{x}|\mathbf{w}] = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_u^2}(\mathbf{w} - \mu_x) + \mu_x \tag{4.6}$$

achieves the minimum mean squared error, where σ_u^2 is the variance of the Gaussian noise $\mathbf{u}$ in (4.3), and we use the convention that adding a scalar to (respectively, subtracting a scalar from) a vector means adding this scalar to (respectively, subtracting this scalar from) every component of the vector. This format in (4.6) is called the Wiener filter in signal processing [120]. It has been shown by Sherman [88, 89] that, besides the ℓ_2-norm error, the linear Wiener filter is also optimal for all ℓ_p-norm errors ($p \geq 1$), including the ℓ_∞-norm error. Surprisingly, we find that, if the input signal is generated by an i.i.d. Gaussian mixture source, then the Wiener filter asymptotically minimizes the expected ℓ_∞-norm error.

Before providing the result for the Gaussian mixture input case, which is mathematically involved, we begin with an analysis of the simpler Bernoulli-Gaussian input case.

Theorem 1 *In parallel Gaussian channels* (4.3)*, if the input signal* $\mathbf{x}$ *is generated by an i.i.d.*

Bernoulli-Gaussian source defined in (4.2)*, then the Wiener filter*

$$\widehat{\mathbf{x}}_{W,BG} = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_u^2}(\mathbf{w} - \mu_x) + \mu_x \tag{4.7}$$

asymptotically achieves the minimum mean ℓ_∞-norm error. More specifically,

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{W,BG}\|_\infty\right]}{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} = 1,$$

where $\widehat{\mathbf{x}}_{\ell_\infty}$ satisfies (4.5)*.*

Theorem 1 is proved in Appendix D. The proof combines concepts in typical sets [28] and a result by Gnedenko [43], which provided asymptotic properties of the maximum of a Gaussian sequence. The main idea of the proof is to show that with overwhelming probability the maximum absolute error satisfies $\|\mathbf{x} - \widehat{\mathbf{x}}\|_\infty = |x_i - \widehat{x}_i|$, where $i \in \mathcal{I} = \{i : x_i \sim \mathcal{N}(\mu_x, \sigma_x^2)\}$, i.e., $\mathcal{I}$ is the index set that includes all the Gaussian components of the vector $\mathbf{x}$, and excludes all the zero components of $\mathbf{x}$. Therefore, minimizing $\|\mathbf{x} - \widehat{\mathbf{x}}\|_\infty$ is equivalent to minimizing $\|\mathbf{x}_\mathcal{I} - \widehat{\mathbf{x}}_\mathcal{I}\|_\infty$, where $(\cdot)_\mathcal{I}$ denotes a subvector with entries in the index set $\mathcal{I}$. Because the vector $\mathbf{x}_\mathcal{I}$ is i.i.d. Gaussian, the Wiener filter minimizes $\|\mathbf{x}_\mathcal{I} - \widehat{\mathbf{x}}_\mathcal{I}\|_\infty$ [88, 89]; hence the Wiener filter minimizes $\|\mathbf{x} - \widehat{\mathbf{x}}\|_\infty$ with overwhelming probability. On the other hand, the cases where $\|\mathbf{x} - \widehat{\mathbf{x}}\|_\infty = |x_i - \widehat{x}_i|$ and $i \notin \mathcal{I}$ are rare, the mean ℓ_∞-norm error of the Wiener filter barely increases, and so the Wiener filter asymptotically minimizes the expected ℓ_∞-norm error.

Having discussed the Bernoulli-Gaussian case, let us proceed to the Gaussian mixture case defined in (4.1). Here the maximum absolute error between $\mathbf{x}$ and the estimate $\widehat{\mathbf{x}}$ satisfies $\|\mathbf{x} - \widehat{\mathbf{x}}\|_\infty = |x_i - \widehat{x}_i|$, where $i \in \mathcal{I}' = \{i : x_i \sim \mathcal{N}(\mu_m, \sigma_m^2)\}$, and $m = \arg\max_{k\in\{1,2,\ldots,K\}} \sigma_k^2$. That is, the maximum absolute error between $\mathbf{x}$ and $\widehat{\mathbf{x}}$ lies in an index that corresponds to the Gaussian mixture component with greatest variance.

Theorem 2 *In parallel Gaussian channels* (4.3)*, if the input signal* $\mathbf{x}$ *is generated by an i.i.d. Gaussian mixture source defined in* (4.1)*, then the Wiener filter*

$$\widehat{\mathbf{x}}_{W,GM} = \frac{\sigma_m^2}{\sigma_m^2 + \sigma_u^2}(\mathbf{w} - \mu_m) + \mu_m \tag{4.8}$$

asymptotically achieves the minimum mean ℓ_∞-norm error, where $m = \arg\max_{k\in\{1,2,\ldots,K\}} \sigma_k^2$. More specifically,

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{W,GM}\|_\infty\right]}{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} = 1,$$

where $\widehat{\mathbf{x}}_{\ell_\infty}$ satisfies (4.5)*.*

The proof of Theorem 2 is given in Appendix E. We note in passing that the statements in

Theorems 1 and 2 do not hold for ℓ_p-norm error ($0 < p < \infty$). Because there is a one to one correspondence between the parameters (μ_k and σ_k^2) of a Gaussian mixture component and its corresponding Wiener filter, if a Wiener filter is optimal in the ℓ_p error sense for any of the Gaussian mixture components, then it is suboptimal for the rest of the mixture components. Therefore, any single Wiener filter is suboptimal in the ℓ_p error sense for any Gaussian mixture signal comprising more than one Gaussian component.

Remark 1 *Theorems 1 and 2 can be extended to compressive signal reconstruction problems. Recall that GAMP and AMP can decouple the compressive sensing system in* (2.11) *and* (2.12) *to an equivalent set of parallel Gaussian channels in* (3.2)*. Therefore, when the input signal* $\mathbf{x}$ *is generated by an i.i.d. Gaussian mixture source, by applying the Wiener filter to* q_j *in* (3.2)*, we can obtain the estimate that minimizes the* ℓ_∞*-norm error of the compressive signal reconstruction process.*

4.5 Practical ℓ_p-Norm Minimization

We have now proved that, when the input signal is i.i.d. Gaussian mixture, the Wiener filter provides asymptotically optimal mean ℓ_∞-norm error in parallel scalar Gaussian channels and linear mixing systems. However, the results are asymptotic in nature, and in practice the ℓ_∞-norm error minimizer must be estimated. In this section, we show some numerical results for a heuristic algorithm that uses an ℓ_p-norm error minimizer to estimate the ℓ_∞-norm error minimizer, where we apply the metric-optimal procedure from Chapter 3. Numerical results show that, with a finite signal length N, the ℓ_p-norm minimizer [100] has lower ℓ_∞-norm error than the Wiener filter.

4.5.1 Parallel scalar Gaussian channels

The channels $w_i = x_i + u_i$ for $i \in \{1, 2, \ldots, N\}$ are separable from each other. If the error metric is additive (3.1) and thus also separable, then estimating the input signal $\mathbf{x}$ from the channel outputs $\mathbf{w}$ can be reduced to scalar reconstruction (estimating x_i from w_i), which is simple to implement in a computationally efficient manner. However, the ℓ_∞-norm error only considers the component with greatest absolute value, and does not have an additive form (3.1). Recall that the definition of the ℓ_p-norm error between $\widehat{\mathbf{x}}$ and $\mathbf{x}$ is

$$\|\widehat{\mathbf{x}} - \mathbf{x}\|_p = \left(\sum_{i \in \{1,\ldots,N\}} |\widehat{x}_i - x_i|^p \right)^{1/p}.$$

This type of error is closely related to our definition of the additive error metric (3.1). We define

$$D_p(\widehat{\mathbf{x}}, \mathbf{x}) = \sum_{i=1}^{N} |\widehat{x}_i - x_i|^p = \|\widehat{\mathbf{x}} - \mathbf{x}\|_p^p, \tag{4.9}$$

and let $\widehat{\mathbf{x}}_p$ denote the estimate that minimizes the conditional expectation of $D_p(\widehat{\mathbf{x}}, \mathbf{x})$, i.e.,

$$\begin{aligned} \widehat{\mathbf{x}}_p &= \arg\min_{\widehat{\mathbf{x}}} E[D_p(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{w}] \\ &= \arg\min_{\widehat{\mathbf{x}}} E[\|\widehat{\mathbf{x}} - \mathbf{x}\|_p^p|\mathbf{w}]. \end{aligned} \tag{4.10}$$

Because scalar channels are separable, equation (4.10) reduces to a scalar reconstruction,

$$\begin{aligned} \widehat{x}_{p,i} &= \arg\min_{\widehat{x}_i} E[|\widehat{x}_i - x_i|^p|w_i] \\ &= \arg\min_{\widehat{x}_i} \int |\widehat{x}_i - x_i|^p f(x_i|w_i) dx_i \end{aligned} \tag{4.11}$$

for $i \in \{1, 2, ..., N\}$, where $f(x_i|w_i)$ can be calculated using Bayes' rule,

$$f(x_i|w_i) = \frac{f(w_i|x_i)f(x_i)}{\int f(w_i|x_i)f(x_i)dx_i}, \tag{4.12}$$

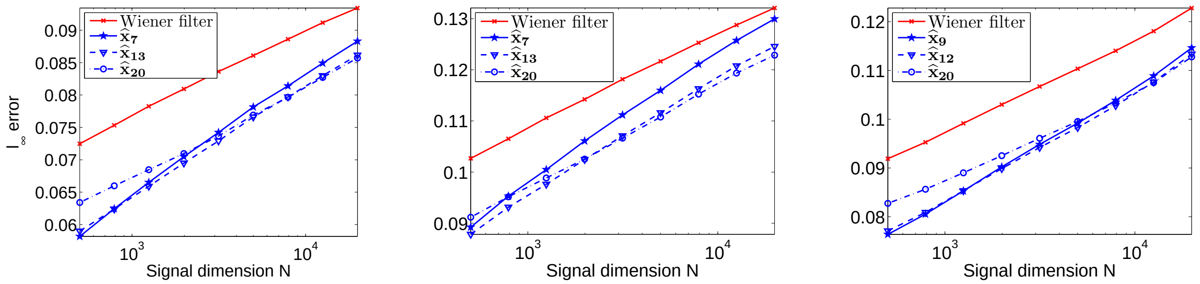
where $f(w_i|x_i)$ is obtained from (4.4). Although $\widehat{\mathbf{x}}_p$ minimizes the $(\ell_p)^p$ error, rather than the ℓ_p-norm error, we call $\widehat{\mathbf{x}}_p$ the ℓ_p-norm minimizer for simplicity. Because it can be shown that

$$\lim_{p\to\infty} \|\widehat{\mathbf{x}} - \mathbf{x}\|_p = \|\widehat{\mathbf{x}} - \mathbf{x}\|_\infty,$$

it is reasonable to expect that if we set p to a large value, then running our ℓ_p-norm minimizer (4.10) will give a solution that converges to an estimate that minimizes the ℓ_∞-norm error.

4.5.2 Compressive sensing systems

Recall from Section 2.2.1 that in the last iteration of GAMP [77], the compressive sensing system in (1.1) is converted to parallel scalar Gaussian channels in (3.2). Therefore, the minimum mean ℓ_p-norm error estimator in (4.10) can be applied to compressive sensing systems. We first compute the conditional probability density function $f_{\mathbf{X}|\mathbf{Q}}(\mathbf{x}|\mathbf{q})$ using Bayes' rule (similar to (4.12)), where $\mathbf{q}$ is obtained from GAMP (2.15). Then, given an additive error metric $D(\widehat{\mathbf{x}}, \mathbf{x})$, the optimal estimate $\widehat{\mathbf{x}}_{\text{opt}}$ is generated by minimizing the conditional expectation of the error



(a) Case 1 (b) Case 2 (c) Case 3

Figure 4.1: *The performance of the Wiener filter and the ℓ_{p_1}, ℓ_{p_2}, and ℓ_{p_3}-norm minimizers in terms of ℓ_∞-norm error in parallel scalar Gaussian channels. The optimal p_{opt} increases as N increases. SNR is 20 dB. (a) $p_1 = 7$, $p_2 = 13$, $p_3 = 20$; (b) $p_1 = 7$, $p_2 = 13$, $p_3 = 20$; and (c) $p_1 = 9$, $p_2 = 12$, $p_3 = 20$.*

metric $E[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}]$:

$$\widehat{\mathbf{x}}_{\text{opt}} = \arg\min_{\widehat{\mathbf{x}}} E[D(\widehat{\mathbf{x}}, \mathbf{x})|\mathbf{q}]. \tag{4.13}$$

In the large system limit, the estimate satisfying (4.13) is asymptotically optimal, because it minimizes the conditional expectation of the error metric. Similar to (4.10) and (4.11), the estimate $\widehat{\mathbf{x}}_{\text{opt}}$ is solved in a component-wise fashion:

$$\widehat{x}_{\text{opt},i} = \arg\min_{\widehat{x}_i} \int D(\widehat{x}_i, x_i) f(x_i|q_i) dx_i, \tag{4.14}$$

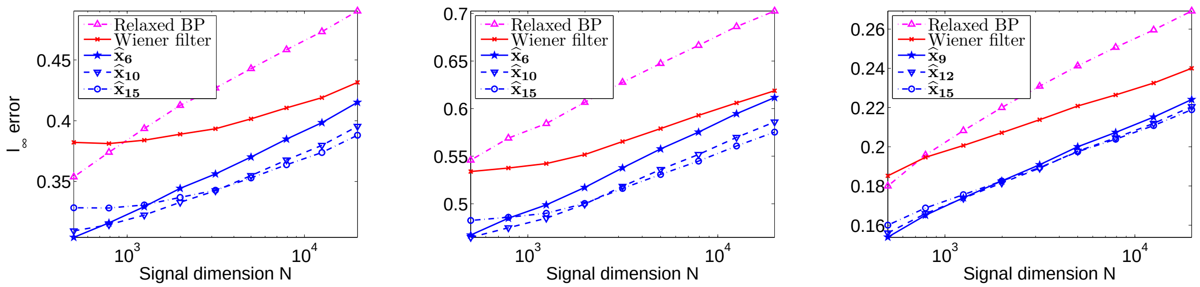
for each x_i, $i \in \{1, 2, \ldots, N\}$. Finally, the ℓ_p-norm minimizer for linear mixing systems is

$$\widehat{x}'_{p,i} = \arg\min_{\widehat{x}_i} \int |\widehat{x}_i - x_i|^p f(x_i|q_i) dx_i \tag{4.15}$$

for $i \in \{1, 2, \ldots, N\}$.

4.6 Numerical Results

Recall that the Wiener filter (4.8) is asymptotically optimal, but its performance for finite N is not clear. On the other hand, the ℓ_p-norm minimizer is a heuristic for finite N. Let us compare the two approaches numerically for both parallel scalar Gaussian channels (2.15) and compressive sensing systems (1.1).



(a) Case 1 (b) Case 2 (c) Case 3

Figure 4.2: *The performance of the Wiener filter and the ℓ_{p_1}, ℓ_{p_2}, and ℓ_{p_3}-norm minimizers in terms of ℓ_∞-norm error in linear mixing systems. The optimal p_{opt} increases as N increases. SNR is 20 dB. (a) $p_1 = 6$, $p_2 = 10$, $p_3 = 15$, $M/N = 0.3$; (b) $p_1 = 6$, $p_2 = 10$, $p_3 = 15$, $M/N = 0.4$; and (c) $p_1 = 9$, $p_2 = 12$, $p_3 = 15$, $M/N = 0.4$.*

4.6.1 Parallel scalar Gaussian channels

We first test for the parallel scalar Gaussian channels $\mathbf{w} = \mathbf{x} + \mathbf{u}$ (4.3), where the input signal $\mathbf{x}$ is generated by the following 3 sources,

- Case 1: i.i.d. sparse Gaussian with sparsity rate 0.05, $x_i \sim 0.05 \cdot \mathcal{N}(0,1) + 0.95 \cdot \delta(x_i)$.
- Case 2: i.i.d. sparse Gaussian with sparsity rate 0.1, $x_i \sim 0.1 \cdot \mathcal{N}(0,1) + 0.9 \cdot \delta(x_i)$.
- Case 3: i.i.d. Gaussian mixture, $x_i \sim 0.01 \cdot \mathcal{N}(0,2) + 0.03 \cdot \mathcal{N}(0,1) + 0.06 \cdot \mathcal{N}(0,0.5) + 0.9 \cdot \delta(x_i)$.

The noise is $u_i \sim \mathcal{N}(0, \sigma_u^2)$, where the noise variance σ_u^2 is set such that the signal to noise ratio (SNR) satisfies

$$\text{SNR} = \frac{E\left[\sum_{i=1}^N x_i^2\right]}{E\left[\sum_{i=1}^N u_i^2\right]} = 100 = 20 \text{ dB}.$$

The Wiener filter is calculated by equation (4.8). For each case, we choose 3 values of p (p_1, p_2, and p_3), and obtain the ℓ_p-norm minimizers by equation (4.10). Figure 4.1 illustrates the ℓ_∞-norm errors for Cases 1–3. In each subfigure, we compare the ℓ_∞-norm errors achieved by the Wiener filter (solid line with cross markers) and by the ℓ_p-norm error minimizers, $\widehat{\mathbf{x}}_{p_1}$ (solid line with pentagram markers), $\widehat{\mathbf{x}}_{p_2}$ (dashed with inverse triangles), and $\widehat{\mathbf{x}}_{p_3}$ (dash-dot with circles). The horizontal axis represents the signal dimension N varying from 500–20,000, and the vertical axis represents the mean ℓ_∞-norm error. The values shown are averages over 15,000 repeated tests.

It is shown in Figure 4.1 that the curves corresponding to the Wiener filter increase more slowly as a function of N than the curves corresponding to the ℓ_p-norm minimizers. Therefore, we can see that the Wiener filter minimizes the mean ℓ_∞-norm error when $N \to \infty$.

At the same time, the ℓ_p-norm error minimizers (4.10) with different values of p indeed outperform the Wiener filter for finite N. For each N, there exists an optimal value of p such that $\widehat{\mathbf{x}}_p$ outperforms all the other estimators; and the optimal p increases as N increases. An intuitive explanation is that as N increases, the probability that larger errors occur also increases, and thus a larger p in (4.9) is used to suppress larger outliers.

Figures 4.1a and 4.1b correspond to different sparsity rates, $s = 0.05$ and 0.1. The figures suggest that the value of p_{opt} for a fixed signal dimension N is related to the sparsity rate of the input signal $\mathbf{x}$.

4.6.2 Compressive sensing systems

We perform simulations for compressive sensing systems (1.1) using the software package "GAMP" [80] and our metric-optimal algorithm in Chapter 3. Our metric-optimal software package can be found at `http://people.engr.ncsu.edu/dzbaron/software/arb_metric/`, and it automatically computes equation (4.14) where the distortion function (3.1) is given as the input of the algorithm.

Again, we test for settings where input signals are generated by the 3 different sources (Cases 1–3 in Section 4.6.1). The measurement matrices $\mathbf{A}$ are i.i.d. zero-mean Gaussian and are normalized to have unit-norm columns. The channels (4.3) are AWGN, and the noise variance is set such that SNR is 20 dB. The number of measurements M in the matrices $\mathbf{A}$ varies as follows: (*i*) in Case 1, $M/N = 0.3$; (*ii*) in Case 2, $M/N = 0.4$; and (*iii*) in Case 3, $M/N = 0.4$. (Cases 2 and 3 are more complicated signals, and so more measurements are necessary for reasonable performance.) As before, the signal dimension N ranges from 500 to $20,000$.

We run the Wiener filter approach (4.8), GAMP [77, 80], and our ℓ_p-norm minimizers with different values of p (4.15). Figure 4.2 shows results for the 3 cases. It can be seen from Figure 4.2 that the Wiener filter outperforms GAMP (dash-dot line with triangle markers) for most N, whereas $\widehat{\mathbf{x}}_p$'s outperform the Wiener filter. Moreover, as N increases, p_{opt} increases.

To summarize, the heuristic approach of the ℓ_p-norm minimizers complements the asymptotic optimality of the Wiener filter. On the other hand, it seems unlikely that this heuristic is truly optimal, and we leave the design of signal reconstruction algorithms with optimal ℓ_∞-norm error for future work.

4.7 Conclusion

In this chapter, we studied the signal reconstruction problem in parallel Gaussian channels and in compressive sensing systems, where the signals are generated by i.i.d. Gaussian mixture sources, and the ℓ_∞-norm error was used to quantify the performance. We proved that in parallel Gaussian channels the Wiener filter (4.8), a simple linear function that is applied to the Gaussian channel outputs, asymptotically minimizes the mean ℓ_∞-norm error when the signal dimension $N \to \infty$. Specifically, the multiplicative constant of the linear filter only relates to the greatest variance of the Gaussian mixture components and the variance of the Gaussian noise.

Our theoretical results in the first half of the chapter are asymptotic, but in the second half of the chapter, we applied the metric-optimal procedure from Chapter 3. The numerical results showed that, with a finite signal length N, the ℓ_p-norm minimizer [100] has lower ℓ_∞-norm error than the Wiener filter.

We also applied the ℓ_p-norm minimizer to compressive sensing systems in settings where the measurement matrix $\mathbf{A}$ is sparse i.i.d. and a compressive sensing system can be decoupled to parallel scalar Gaussian channels [46, 47, 105]. Let us highlight that the channels (1.1) in the compressive sensing systems are not restricted to Gaussian. Again, for a finite N, the ℓ_p-norm minimizer outperforms the Wiener filter for ℓ_∞-norm error.

Chapter 5

Compressive Imaging

Having studied theoretical properties of compressive signal reconstruction with arbitrary error metrics, we further explore practical compressive imaging problems, where the input signal is a vectorized image, and the goal is to acquire the image using as few measurements as possible. Acquiring images in a compressive manner requires less sampling time than conventional imaging technologies. Applications of compressive imaging appear in medical imaging [16, 61, 70], seismic imaging [25], and hyperspectral imaging [74, 121].

In this chapter, we propose compressive imaging algorithms, where we employ two wavelet based image denoisers within the approximate message passing (AMP) framework. The proposed algorithms in this chapter were first described in Tan et al. [101, 102]. We will continue our discussion about compressive imaging problems in Chapter 6, where compressive hyperspectral imaging problems are considered.

5.1 Related Work

Many compressive imaging algorithms have been proposed in the literature. For example, Som and Schniter [91] modeled the structure of the wavelet coefficients by a hidden Markov tree (HMT), and applied a turbo scheme that alternates between inference on the HMT structure with standard belief propagation and inference on the compressive sensing measurement structure with the generalized approximate message passing algorithm. He and Carin [51] proposed a hierarchical Bayesian approach with Markov chain Monte Carlo (MCMC) for natural image reconstruction. Soni and Haupt [92] exploited a hierarchical dictionary learning method [56] and assumed that projecting images onto the learned dictionary will yield tree-sparsity, and therefore the nonzero supports of the dictionary can be identified and estimated accurately by setting an appropriate threshold.

However, existing compressive imaging algorithms may either not achieve good reconstruction quality or not be fast enough. Therefore, in this chapter, we focus on a variation of a fast and effective algorithm called approximate message passing (AMP) [33] to improve over the prior art. AMP is an iterative signal reconstruction algorithm that performs scalar denoising within each iteration, and proper selection of the denoising function used within AMP is needed to obtain better reconstruction quality. One challenge in applying image denoisers within AMP is that it may be hard to compute the so-called "Onsager reaction term" in (2.17) [33, 107] in the AMP iteration steps. The Onsager reaction term includes the derivative of the image denoising function, and thus if an image function does not have a convenient closed form, then the Onsager reaction term may be difficult to compute.

Dictionary learning is an effective technique that has attracted a great deal of attention in image denoising. Dictionary learning based methods [36, 123] generally achieve lower reconstruction error than wavelet-based methods. However, the learning procedure requires a large amount of training images, and may involve manual tuning. Owing to these limitations, our main focus in this chapter is to integrate relatively simple and fast image denoisers into compressive imaging reconstruction algorithms.

Dabov et al. [29] developed an image denoising strategy that employs collaborative filtering in a sparse 3-D transform domain, and they offered an efficient implementation that achieves favorable denoising quality. Other efficient denoising schemes include wavelet-based methods. A typical wavelet-based image denoiser proceeds as follows: (*i*) apply a wavelet transform to the image and obtain wavelet coefficients; (*ii*) denoise the wavelet coefficients; and (*iii*) apply an inverse wavelet transform to the denoised wavelet coefficients, yielding a denoised image. Two popular examples of denoisers that can be applied to the wavelet coefficients are hard thresholding and soft thresholding [32]. Variations on the thresholding scheme can be found in [23, 37]; other wavelet-based methods were proposed by Simoncelli and Adelson [90], Mıhçak et al. [58], and Moulin and Liu [68].

5.2 Image Reconstruction Algorithm

In this chapter, we focus on the system model defined in (1.1) where both the measurement matrix $\mathbf{A}$ and the noise $\mathbf{z}$ are independent and identically distributed (i.i.d.) Gaussian. Recall that in Chapter 2 we introduced the AMP algorithm. In the t-th iteration, we obtain the vectors $\mathbf{x}^t \in \mathbb{R}^N$ in (2.16) and $\mathbf{r}^t \in \mathbb{R}^M$ in (2.17). The vector $\mathbf{A}^T\mathbf{r}^t + \mathbf{x}^t \in \mathbb{R}^N$ in (2.16) can be regarded as noisy measurements of $\mathbf{x}$ in the t-th iteration with noise variance σ_t^2, and therefore the denoising function $\eta_t(\cdot)$ is performed on a scalar channel. Let us denote the equivalent scalar channel at iteration t by

$$\mathbf{q}^t = \mathbf{A}^T\mathbf{r}^t + \mathbf{x}^t = \mathbf{x} + \mathbf{v}^t, \tag{5.1}$$

where $v_i^t \sim \mathcal{N}(0, \sigma_t^2)$. The asymptotic performance of AMP can be characterized by a state evolution (SE) formalism:

$$\sigma_{t+1}^2 = \sigma_z^2 + \frac{1}{R} E\left[\left(\eta_t\left(X + \sigma_t W\right) - X\right)^2\right], \tag{5.2}$$

where the random variables $W \sim \mathcal{N}(0,1)$ and $X \sim f_X$. Formal statements about SE appear in Bayati and Montanari [10]. Note that SE (5.2) tracks the noise variance for AMP iterations, but the noise variance σ_{t+1}^2 need not necessarily be the smallest possible, unless in each iteration the denoiser $\eta_t(\cdot)$ achieves the minimum mean square error (MMSE). On the other hand, it is unrealistic to expect existing image denoisers to achieve the MMSE, because the statistical distribution of natural images has yet to be determined. That said, running AMP with good image denoisers that achieve lower mean square error (MSE) may yield lower MSE in compressive imaging problems.

Finally, SE theoretically characterizes the noise variance σ_t^2 of the scalar channel at each iteration. However, the MSE performance of image denoisers cannot be characterized theoretically. Therefore, we must estimate the effective Gaussian noise level σ_t^2 empirically in each AMP iteration. The estimated noise variance $\widehat{\sigma}_t^2$ can be calculated as [66]:

$$\widehat{\sigma}_t^2 = \frac{1}{M} \sum_{i=1}^{M} (r_i^t)^2, \tag{5.3}$$

where $\mathbf{r}^t$ is defined in (2.17).

Before we end this section, let us highlight that in Chapters 3 and 4, we utilized the parallel scalar Gaussian channel in the last iteration of GAMP, while in the compressive imaging algorithm we describe in this chapter, image denoisers are applied to the parallel scalar Gaussian channels in every iteration.

5.2.1 Wavelet transforms in AMP

In this section, we describe how wavelet-based image denoisers are applied within AMP, and then outline two image denoisers that were proposed by Figueiredo and Nowak [37] and Mıhçak et al. [58], respectively.

In image processing, one often computes the wavelet coefficients [63] of images, applies some signal processing technique to the wavelet coefficients, and finally applies the inverse wavelet transform to the processed coefficients to obtain processed images. We now show how image denoising can be performed within AMP in the wavelet domain. Let us denote the wavelet transform by $\mathcal{W}$ and the inverse wavelet transform by $\mathcal{W}^{-1}$. By applying the wavelet transform to a vectorized image signal $\mathbf{x}$ (a 2-dimensional wavelet transform is used), we obtain the wavelet coefficient vector $\theta_{\mathbf{x}} = \mathcal{W}\mathbf{x}$. Conversely, $\mathbf{x} = \mathcal{W}^{-1}\theta_{\mathbf{x}}$. Therefore, the matrix channel (1.1) becomes $\mathbf{y} = \mathbf{A}\mathcal{W}^{-1}\theta_{\mathbf{x}} + \mathbf{z}$, where $\mathbf{A}\mathcal{W}^{-1}$ can be regarded as a new matrix in the matrix channel (1.1) and $\theta_{\mathbf{x}}$ as the corresponding input signal.

Let us express the AMP iterations (2.16, 2.17) for settings where the matrix is $\mathbf{A}\mathcal{W}^{-1}$,

$$\begin{aligned}
\theta_{\mathbf{x}}^{t+1} &= \eta_t((\mathbf{A}\mathcal{W}^{-1})^T\mathbf{r}^t + \theta_{\mathbf{x}}^t), && (5.4)\\
\mathbf{r}^t &= \mathbf{y} - (\mathbf{A}\mathcal{W}^{-1})\theta_{\mathbf{x}}^t \\
&\quad + \frac{1}{R}\mathbf{r}^{t-1}\langle\eta_{t-1}'((\mathbf{A}\mathcal{W}^{-1})^T\mathbf{r}^{t-1} + \theta_{\mathbf{x}}^{t-1})\rangle \\
&= \mathbf{y} - \mathbf{A}\mathbf{x}^t \\
&\quad + \frac{1}{R}\mathbf{r}^{t-1}\langle\eta_{t-1}'((\mathbf{A}\mathcal{W}^{-1})^T\mathbf{r}^{t-1} + \theta_{\mathbf{x}}^{t-1})\rangle. && (5.5)
\end{aligned}$$

Because the wavelet transform matrix is orthonormal, i.e., $\mathcal{W}\mathcal{W}^T = \mathbf{I} = \mathcal{W}\mathcal{W}^{-1}$, it can be shown that $(\mathbf{A}\mathcal{W}^{-1})^T = \mathcal{W}\mathbf{A}^T$. Therefore, the input of the denoiser $\eta_t(\cdot)$ (5.4) becomes

$$(\mathbf{A}\mathcal{W}^{-1})^T\mathbf{r}^t + \theta_{\mathbf{x}}^t = \mathcal{W}\mathbf{A}^T\mathbf{r}^t + \theta_{\mathbf{x}}^t = \mathcal{W}\mathbf{A}^T\mathbf{r}^t + \mathcal{W}\mathbf{x}^t = \mathcal{W}\mathbf{q}^t, \qquad (5.6)$$

where $\mathbf{q}^t$ (6.5) is the noisy image at iteration t, and $\mathcal{W}\mathbf{q}^t$ is the wavelet transform applied to the noisy image.

With the above analysis of the modified AMP (5.4, 5.5), we formulate a compressive imaging procedure as follows. Let us denote the the wavelet transform of the scalar channel (6.5) by

$$\theta_{\mathbf{q}}^t = \theta_{\mathbf{x}} + \theta_{\mathbf{v}}^t, \qquad (5.7)$$

where $\theta_{\mathbf{q}}^t = \mathcal{W}\mathbf{q}^t$, $\theta_{\mathbf{x}} = \mathcal{W}\mathbf{x}$, and $\theta_{\mathbf{v}}^t = \mathcal{W}\mathbf{v}^t$. First, $\mathbf{r}^t$ and $\mathbf{x}^t$ are initialized to all-zero vectors. Then, at iteration t the algorithm proceeds as follows,

1. Calculate the residual term $\mathbf{r}^t$.
2. Calculate the noisy image $\mathbf{q}^t = \mathbf{A}^T\mathbf{r}^t + \mathbf{x}^t$, and apply the wavelet transform $\mathcal{W}$ to the noisy image $\mathbf{q}^t$ to obtain wavelet coefficients $\theta_{\mathbf{q}}^t$, which are the inputs of the scalar denoiser $\eta_t(\cdot)$ in (5.4).
3. Apply the denoiser $\eta_t(\cdot)$ to the wavelet coefficients $\theta_{\mathbf{q}}^t$, and obtain denoised coefficients $\theta_{\mathbf{x}}^{t+1}$.
4. Apply the inverse wavelet transform $\mathcal{W}^{-1}$ to the coefficients $\theta_{\mathbf{x}}^{t+1}$ to obtain the estimated image $\mathbf{x}^{t+1}$, which is used to compute the residual term in the next iteration.

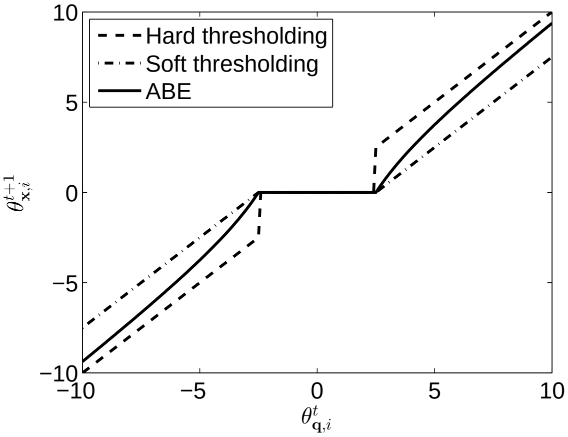


Figure 5.1: *Comparison of ABE with hard and soft thresholding.*

5.2.2 Image denoisers

We choose to denoise the wavelet coefficients using scalar denoisers proposed by Figueiredo and Nowak [37] and Mıhçak et al. [58], respectively, because these two denoisers are simple to implement while revealing promising numerical results (see Section 6.4). We call the algorithm where ABE [37] is utilized within AMP "AMP-ABE," and the algorithm where the adaptive Wiener filter [58] is utilized "AMP-Wiener." In both algorithms, the variance of the noise σ_t^2 in the noisy image is obtained using (5.3). Because we use an orthonormal wavelet transform, the noise variance in the wavelet domain is equal to that in the image domain. Although we only show how to employ two image denoisers within AMP, they serve as a proof of concept that other image denoisers could also be applied within AMP, possibly leading to further improvements in both image reconstruction quality and runtime.

Amplitude-scale-invariant Bayes estimator

Figueiredo and Nowak's denoiser [37] is an amplitude-scale-invariant Bayes estimator (ABE), and it is a scalar function. More specifically, for each noisy wavelet coefficient $\theta^t_{\mathbf{q},i}$ (5.7), the estimate of $\theta_{\mathbf{x},i}$ for the next iteration is

$$\theta^{t+1}_{\mathbf{x},i} = \eta_t(\theta^t_{\mathbf{q},i}) = \frac{\left((\theta^t_{\mathbf{q},i})^2 - 3\sigma_t^2\right)_+}{\theta^t_{\mathbf{q},i}}, \tag{5.8}$$

where σ_t^2 is the noise variance of the scalar channel (6.5) at the t-th AMP iteration, and $(\cdot)_+$ is a function such that $(u)_+ = u$ if $u > 0$ and $(u)_+ = 0$ if $u \leq 0$. Note that because the wavelet transform matrix $\mathcal{W}$ is orthonormal, the variance of the noise $\mathbf{v}^t$ (6.5) is equal to the variance of $\theta_{\mathbf{v}}^t$ (5.7).

Figure 5.1 illustrates the ABE function. It can be seen that ABE offers a compromise between the hard thresholding, i.e., $\eta(u,\tau) = u \cdot \mathbb{1}_{(u>|\tau|)}$, where $\mathbb{1}_{\{\cdot\}}$ denotes an indicator function, and soft thresholding, i.e., $\eta(u,\tau) = \text{sign}(u) \cdot (|u| - \tau)_+$, proposed by Donoho and Johnstone [32]. ABE is convenient to utilize, because there is no need to tune the thresholding values,[1] and ABE has been shown to outperform both hard and soft thresholding methods for image denoising [37].

The ABE function is continuous and differentiable except for two points ($\theta_{\mathbf{q},i}^t = \pm\sqrt{3}\sigma_t$), and we calculate the derivative of this denoising function numerically to obtain the Onsager reaction term in (2.17).

Adaptive Wiener filter

Mıhçak et al. [58] proposed a method to estimate the variances of the wavelet coefficients, and then apply the corresponding Wiener filter to each wavelet coefficient. The variance of the noisy wavelet coefficient $\theta_{\mathbf{q},i}^t$ is estimated from its neighboring coefficients. More specifically, a set of 3×3 or 5×5 neighboring coefficients $\mathcal{N}_i$ that is centered at $\theta_{\mathbf{q},i}^t$ is considered, and the variance of $\theta_{\mathbf{q},i}^t$ is estimated by averaging the sum of $(\theta_{\mathbf{q},k}^t)^2$ where $k \in \mathcal{N}_i$. This method of averaging the neighboring coefficients can be regarded as first convolving a 3×3 or 5×5 mask of all 1's with the matrix of squared wavelet coefficients $\theta_{\mathbf{q}}^t$, and then dividing by the normalizing constant 9 (for a 3×3 mask) or 25 (for a 5×5 mask). Other masks can be applied to produce different and possibly better denoising results. For example, we have found that the mask

$$\begin{array}{ccccc} & 1 & 1 & 1 & \\ 1 & 1 & 2 & 1 & 1 \\ 1 & 2 & 3 & 2 & 1 \\ 1 & 1 & 2 & 1 & 1 \\ & 1 & 1 & 1 & \end{array}$$

obtains lower MSE than other 5×5 masks we have considered. Recall the scalar channel defined in (5.7) where the noise variance is σ_t^2; we estimate the variance of a noisy wavelet coefficient $\theta_{\mathbf{q},i}^t$ by $\widehat{\sigma}_i^2$, and the variance of the true wavelet coefficient $\theta_{\mathbf{x},i}^t$ by $\widehat{\sigma}_i^2 - \sigma_t^2$.[2] Therefore, the scaling factor in the Wiener filter [120] is given by $\frac{\widehat{\sigma}_i^2 - \sigma_t^2}{(\widehat{\sigma}_i^2 - \sigma_t^2) + \sigma_t^2}$, and the adaptive Wiener filter being used as the denoising function can be expressed as follows,

$$\theta_{\mathbf{x},i}^{t+1} = \eta_t(\theta_{\mathbf{q},i}^t) = \frac{\widehat{\sigma}_i^2 - \sigma_t^2}{(\widehat{\sigma}_i^2 - \sigma_t^2) + \sigma_t^2}\theta_{\mathbf{q},i}^t = \frac{\widehat{\sigma}_i^2 - \sigma_t^2}{\widehat{\sigma}_i^2}\theta_{\mathbf{q},i}^t. \tag{5.9}$$

[1] We note in passing that Mousavi et al. [69] proposed to tune the soft threshold automatically.

[2] We use $\max\{\widehat{\sigma}_i^2 - \sigma_t^2, 0\}$ to restrict the variance to be non-negative.

Finally, the derivative of this denoising function with respect to $\theta^t_{\mathbf{q},i}$ is simply the scaling factor $\frac{\widehat{\sigma}_i^2-\sigma_t^2}{\widehat{\sigma}_i^2}$ of the Wiener filter, and so the Onsager reaction term in (2.17) can be obtained efficiently.

In standard AMP [33], the denoising function $\eta_t(\cdot)$ is separable, meaning that $\theta^{t+1}_{\mathbf{x},i}$ only depends on its corresponding noisy wavelet coefficient $\theta^t_{\mathbf{q},i}$. In the adaptive Wiener filter, however, the estimated variance $\widehat{\sigma}_i^2$ of each noisy wavelet coefficient depends on the neighboring coefficients of $\theta^t_{\mathbf{q},i}$, and so the denoising function in (5.9) implicitly depends on the neighboring coefficients of $\theta^t_{\mathbf{q},i}$. Therefore, the adaptive Wiener filter in (5.9) is not a strictly separable denoising function, and AMP-Wiener encounters convergence issues. Fortunately, a technique called "damping" [84] solves for the convergence problem of AMP-Wiener. Specifically, damping is an extra step in the AMP iteration (2.16); instead of updating the value of $\mathbf{x}^{t+1}$ by the output of the denoiser $\eta_t(\mathbf{A}^T\mathbf{r}^t+\mathbf{x}^t)$, we assign a weighted sum of $\eta_t(\mathbf{A}^T\mathbf{r}^t+\mathbf{x}^t)$ and $\mathbf{x}^t$ to $\mathbf{x}^{t+1}$ as follows,

$$\mathbf{x}^{t+1} = (1-\lambda)\cdot\eta_t(\mathbf{A}^T\mathbf{r}^t+\mathbf{x}^t)+\lambda\cdot\mathbf{x}^t, \tag{5.10}$$

for some constant $0 \leq \lambda < 1$. It has been shown by Rangan et al. [84] that sufficient damping ensures the convergence of AMP where the measurement matrix $\mathbf{A}$ is not i.i.d. Gaussian. However, we did indeed use i.i.d. Gaussian matrices in our numerical results in Section 6.4, and damping solved the convergence problem of AMP-Wiener, which suggests that damping may be an effective technique when various convergence issues arise in AMP based algorithms. We note in passing that other techniques such as SwAMP [64] and ADMM-GAMP [82] also solve for the convergence problem in AMP.

5.3 Numerical Results

Having described the AMP algorithm [33] and two image denoisers [37, 58], in this section we present the numerical results of applying these two denoisers within AMP.

5.3.1 Reconstruction quality and runtime

We compare AMP-ABE and AMP-Wiener with three prior art compressive imaging algorithms, (*i*) Turbo-BG proposed by Som and Schniter [91]; (*ii*) Turbo-GM, also by Som and Schniter [91]; and (*iii*) a Markov chain Monte Carlo (MCMC) method by He and Carin [51]. Both Turbo-BG and Turbo-GM are also message passing based algorithms. However, these two algorithms require more computation than AMP-ABE and AMP-Wiener, because they include two message passing procedures; the first procedure solves for dependencies between the wavelet coefficients and the second procedure is AMP. The performance metrics that we use to compare the algorithms are runtime and normalized MSE (NMSE), $\text{NMSE}(\mathbf{x},\widehat{\mathbf{x}}) = 10\log_{10}(\|\mathbf{x}-\widehat{\mathbf{x}}\|_2^2/\|\mathbf{x}\|_2^2)$, where $\widehat{\mathbf{x}}$ is the estimate of the vectorized input image $\mathbf{x}$. In all simulations, we use the Haar wavelet transform [63].

Let us begin by contrasting the three prior art compressive imaging algorithms based on the numerical results provided in [91]. Turbo-BG and Turbo-GM have similar runtimes; the NMSE of Turbo-GM is typically 0.5 dB better (lower) than the NMSE of Turbo-BG. At the same time, the NMSE of the MCMC algorithm [51] is comparable to those of Turbo-BG and Turbo-GM, but MCMC is 30 times slower than the Turbo approaches of Som and Schniter [91]. Other algorithms have also been considered for compressive imaging. For example, compressive sampling matching pursuit (CoSaMP) [71] requires only half the runtime of Turbo-GM, but its NMSE is roughly 4 dB worse than that of Turbo-GM; and model based CS [8] is twice slower than Turbo-GM and its NMSE is also roughly 4 dB worse. Therefore, we provide numerical results for Turbo-BG, Turbo-GM, MCMC, and our two proposed AMP based approaches.

Numerical setting: We downloaded 591 images from "pixel-wise labeled image database v2" at http://research. microsoft.com/en-us/projects/objectclassrecognition, and extracted image patches using the following two methods.

- Method 1: A 192 × 192 patch is extracted from the upper left corner of each image, and then the patch is resized to 128 × 128; this image patch extraction method was used by Som and Schniter [91].
- Method 2: A 192×192 patch is extracted from the upper left corner of each image without resizing.

The measurement matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$ is generated with i.i.d. Gaussian entries distributed as $\mathcal{N}(0, \frac{1}{M})$; each column is then normalized to have unit norm. For 128 × 128 patches extracted by Method 1, the number of measurements $M = 5,000$, which is identical to the numerical setting by Som and Schniter [91]. For 192 × 192 patches extracted by Method 2, the number of measurements $M = 11,059$, i.e., the measurement rate is 0.3. In both methods, the measurements $\mathbf{y}$ are noiseless, i.e., $\mathbf{y} = \mathbf{Ax}$. Finally, we set the number of AMP iterations to be 30 and the damping constant λ (5.10) for AMP-Wiener to be 0.1.

Result 1: Tables 5.1 and 5.2 show the NMSE and runtime averaged over the 591 image patches that are extracted by Methods 1 and 2, respectively. Runtime is measured in seconds on a Dell OPTIPLEX 9010 running an Intel(R) Core$^{\text{TM}}$ i7-860 with 16GB RAM, and the environment is Matlab R2013a. Figures 5.2 and 5.3 complement Tables 5.1 and 5.2, respectively, by plotting the average NMSE over 591 images from iteration 1 to iteration 30.

It can be seen from Table 5.1 that the NMSE of AMP-Wiener is the best (lowest) among all the algorithms compared. At the same time, AMP-Wiener runs approximately 3.5 times faster than the Turbo approaches of Som and Schniter [91], and 120 times faster than MCMC [51].

Algorithm	NMSE (dB)	Runtime (sec)
Turbo-BG [91]	-20.37	12.39
Turbo-GM [91]	-20.72	12.47
MCMC [51]	-20.31	423.15
AMP-ABE	-19.30	3.39
AMP-Wiener	-21.00	3.34

Table 5.1: *NMSE and runtime averaged over 591 image patches: a* 192×192 *patch from the upper left corner of each image is first extracted, and then resized to* 128×128*. The number of measurements* $M = 5,000$*, and the measurements are noiseless.*

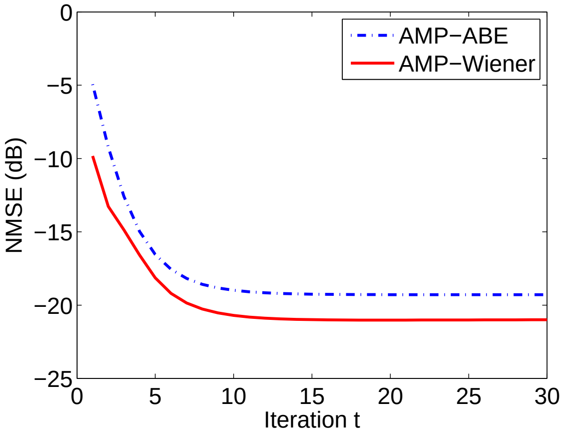


Figure 5.2: *Average NMSE over 591 images from AMP iteration 1 to iteration 30. Image patches are extracted by Method 1: a* 192×192 *patch from the upper left corner of each image is first extracted, and then resized to* 128×128*.*

Although AMP-ABE does not outperform the competing algorithms in terms of NMSE, it runs as fast as AMP-Wiener.

Table 5.1 presents the runtimes of AMP-ABE and AMP-Wiener for image patches extracted by Method 1 with 30 iterations. However, we can see from Figure 5.2 that AMP-ABE and AMP-Wiener with fewer iterations already achieve NMSEs that are close to the NMSE shown in Table 5.1. In Figure 5.2, the horizontal axis represents iteration numbers, and the vertical axis represents NMSE. It is shown in Figure 5.2 that the NMSE drops markedly from -10 dB to -21 dB for AMP-Wiener (solid line) and from -5 dB to -19 dB for AMP-ABE (dash-dot line), respectively. Note that the average NMSE is approximately -21 dB for AMP-Wiener and -19 dB for AMP-ABE around iteration 15. Therefore, we may halve the runtimes of AMP-ABE and AMP-Wiener (to approximately 1.7 seconds) by reducing the number of AMP iterations from 30 to 15.

The simulation for the larger image patches extracted by Method 2 is slow, and thus the

Algorithm	NMSE (dB)	Runtime (sec)
Turbo-GM [91]	-19.64	56.12
AMP-ABE	-17.57	15.99
AMP-Wiener	-20.29	15.53

Table 5.2: *NMSE and runtime averaged over 591 images extracted by Method 2: a* 192×192 *patch is extracted from the upper left corner of each image. The number of measurements* $M = 11,059$*, and the measurements are noiseless.*

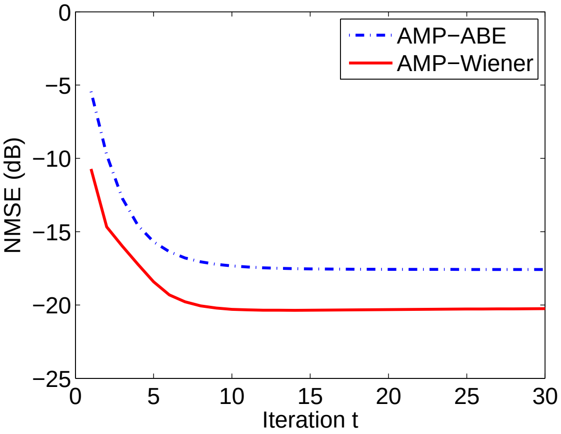


Figure 5.3: *Average NMSE over 591 images from AMP iteration 1 to iteration 30. Image patches are extracted by Method 2: a* 192×192 *patch is extracted from the upper left corner of each image.*

results for Turbo-BG and MCMC have not been obtained for Table 5.2. We believe that Turbo-BG is only slightly worse than Turbo-GM. At the same time, we did test for MCMC on several images, and found that the NMSEs obtained by MCMC were usually 0.5 dB higher than AMP-Wiener and the runtimes of MCMC usually exceeded 1,500 seconds. Similar to Figure 5.2, it can be seen from Figure 5.3 that the runtimes of our AMP based approaches could be further reduced by reducing the number of AMP iterations without much deterioration in estimation quality.

Result 2: As a specific example, Figure 5.4 illustrates one of the 591 image patches and the estimated patches using AMP-Wiener at iterations 1, 3, 7, 15, and 30. We also present the estimated patches using Turbo-GM and MCMC. It can be seen from Figure 5.4 that the estimated images using AMP-Wiener are gradually denoised as the number of iterations is increased, and the NMSE achieved by AMP-Wiener at iteration 15 already produces better reconstruction quality than Turbo-GM and MCMC.

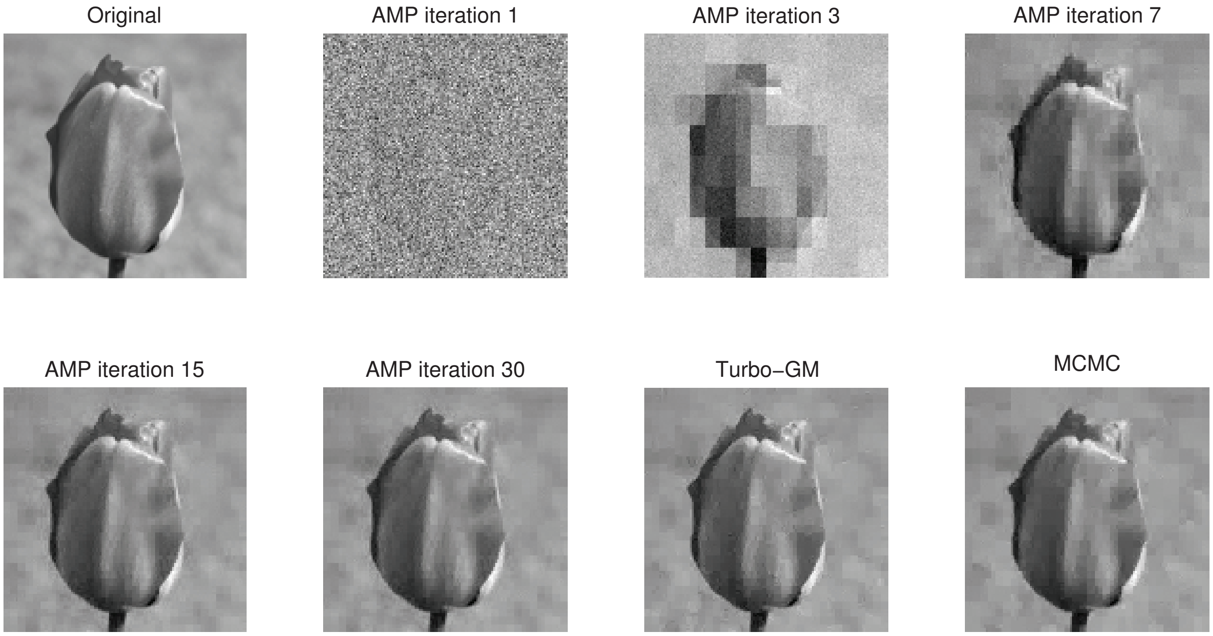


Figure 5.4: *Original "10_5_s.bmp" input image, the estimated images using AMP-Wiener at iterations 1, 3, 7, 15, 30, and the estimated images using Turbo-GM and MCMC. The image patch is extracted by Method 1: a* 192×192 *patch is first extracted, and then resized to* 128×128*. The NMSE of each estimated image is as follows, AMP-Wiener iteration 1,* -11.46 *dB; AMP-Wiener iteration 3,* -18.17 *dB; AMP-Wiener iteration 7,* -26.28 *dB; AMP-Wiener iteration 15,* -30.07 *dB; AMP-Wiener iteration 30,* -30.36 *dB; Turbo-GM [91],* -29.62 *dB; and MCMC [51],* -29.13 *dB.*

5.3.2 Performance of scalar denoisers

Having seen that AMP-Wiener consistently outperforms AMP-ABE, let us now understand why AMP-Wiener achieves lower NMSE than AMP-ABE.

We test for ABE [37] and the adaptive Wiener filter [58] as scalar image denoisers in scalar channels as defined in (6.5). In this simulation, we use the 591 image patches extracted by Method 2, and add i.i.d. Gaussian noise $\mathcal{N}(0, \sigma^2)$ to the image patches. The pixel values are normalized to be between 0 and 1, and we verified from the simulations for Table 5.2 that the estimated noise variances of the scalar channels in AMP iterations are typically between 1×10^{-4} and 1. In Figure 5.5, the vertical axis represents NMSE, and the horizontal axis represents different noise variances varying from 1×10^{-4} to 1 . It is shown in Figure 5.5 that the adaptive Wiener filter (solid line) consistently achieves lower NMSE than ABE (dash-dot line) for all noise variances, which suggests that AMP-Wiener outperforms AMP-ABE in every AMP iteration, and thus outperforms AMP-ABE when we stop iterating in iteration 30. Therefore, in order to achieve favorable reconstruction quality, it is important to select a good

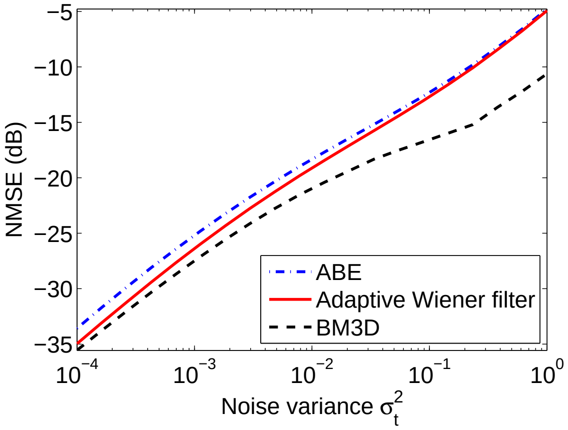


Figure 5.5: *Average NMSE over 591 images versus noise variance. Image patches are extracted by Method 2: a* 192×192 *patch is extracted from the upper left corner of each image. These image denoisers are applied to scalar channels* (6.5).

image denoiser within AMP. With this in mind, we include the NMSE of the image denoiser "block-matching and 3-D filtering" BM3D [29] in Figure 5.5, and find that BM3D (dashed line) has lower NMSE than the adaptive Wiener filter, especially when the noise variance is large. Note that the NMSEs of ABE for different noise variances are within 1 dB of the NMSEs of the adaptive Wiener filter, but this performance gap in scalar denoisers is amplified to more than 2 dB (refer to Table 5.2) when applying the scalar denoisers within AMP. In other words, it is possible that applying BM3D within AMP could achieve better reconstruction quality than AMP-Wiener. However, one challenge of applying BM3D within AMP will be that it is not clear whether the Onsager reaction term in (2.17) can be computed in closed form or numerically, and thus an alternative way of approximating the Onsager reaction term may need to be developed. Note that Metzler et al. [65] recently showed how to approximate the Onsager correction term numerically, thus allowing to use different image denoisers within AMP.

5.3.3 Reconstruction quality versus measurement rate

Finally, we also evaluate the performance of each algorithm by plotting the NMSE (average NMSE over 591 images) versus the measurement rate $R = M/N$. The measurement matrix $\mathbf{A}$ is generated the same way as the numerical setting in Section 5.3.1.

Result: Figures 5.6 and 5.7 illustrate how the NMSEs achieved by AMP-Wiener and Turbo-GM vary when the measurement rate R changes, where the horizontal axis represents the measurement rate $R = M/N$, and the vertical axis represents NMSE. Figures 5.6 shows the results for image patches extracted by Method 1, and the measurement rate R varies from 0.1 to

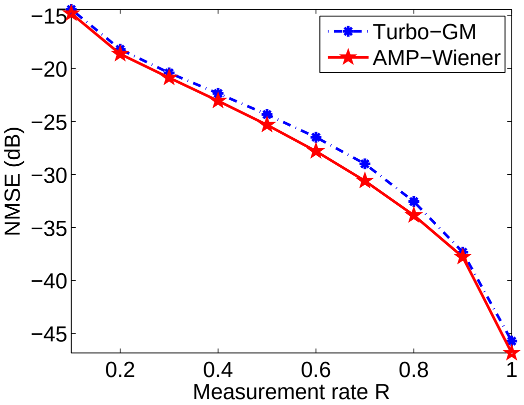


Figure 5.6: *Average NMSE over 591 images versus measurement rate. Image patches are extracted by Method 1: a* 192×192 *patch from the upper left corner of each image is first extracted, and then resized to* 128×128.

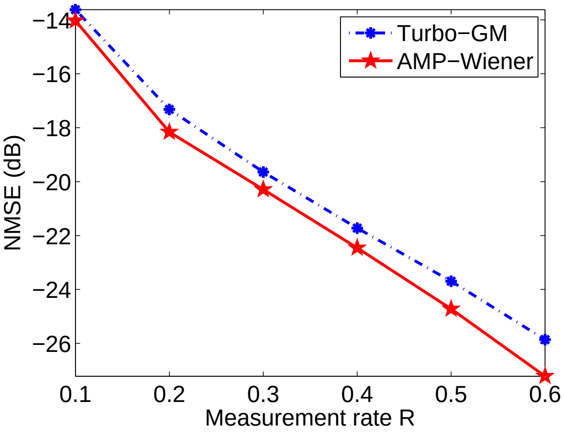


Figure 5.7: *Average NMSE over 591 images versus measurement rate. Image patches are extracted by Method 2: a* 192×192 *patch is extracted from the upper left corner of each image.*

1. Figure 5.7 shows the results for image patches extracted by Method 2. Because the simulation for 192×192 image patches is relatively slow, we only show results for R that varies from 0.1 to 0.6. It can be seen from Figures 5.6 and 5.7 that AMP-Wiener (solid line with pentagram markers) achieves lower NMSE than that of Turbo-GM (dash-dot line with asterisks) for all values of R.

5.4 Conclusion

In this chapter, we proposed compressive imaging algorithms that apply image denoisers within AMP. Specifically, we used the "amplitude-scale-invariant Bayes estimator" (ABE) [37] and an adaptive Wiener filter [58] within AMP. Numerical results showed that AMP-Wiener achieves the lowest reconstruction error among all competing algorithms in all simulation settings, while AMP-ABE also offers competitive performance. Moreover, the runtimes of AMP-ABE and AMP-Wiener are significantly lower than those of MCMC [51] and the Turbo approaches [91], and Figures 5.2 and 5.3 suggested that the runtimes of our AMP based algorithms could be reduced further if we accept a slight deterioration in NMSE.

Recall that the input of the denoising function η_t in (2.16) is a noisy image with i.i.d. Gaussian noise, and so we believe that any image denoiser that deals with i.i.d. Gaussian noise can be applied within AMP. At the same time, in order to develop fast AMP based compressive imaging algorithms, the image denoisers that are applied within AMP should be fast. By comparing the denoising quality of ABE [37] and the adaptive Wiener filter [58] as image denoisers in scalar channels, we have seen that AMP with a better denoiser produces better reconstruction quality for compressive imaging problems. With this in mind, employing more advanced image denoisers within AMP may produce promising results for compressive imaging problems.

Chapter 6

Compressive Hyperspectral Imaging

Having investigated the compressive imaging problem with two-dimensional images, we further study a specific application of compressive imaging, compressive hyperspectral imaging, in this chapter. We modify the AMP-Wiener algorithm that is described in Chapter 5 and apply the algorithm to three-dimensional (3D) hyperspectral image reconstruction. We call the modified algorithm AMP-3D-Wiener. The algorithm proposed in this chapter appeared in Tan et al. [103, 104].

A hyperspectral image is a 3D image cube comprised of a collection of two-dimensional (2D) images (slices), where each 2D image is captured at a specific wavelength. Hyperspectral images allow us to analyze spectral information about each spatial point in a scene, and thus can help us identify different materials that appear in the scene [53]. Therefore, hyperspectral imaging has applications to areas such as medical imaging [72, 87], remote sensing [57], geology [59], and astronomy [52].

One conventional approach to acquire hyperspectral images is to use a spectrometer, which captures spatial information sequentially. However, spectrometers have some disadvantages: (i) data acquisition takes a long time, because the scanning is sequential; and (ii) large amounts of data are acquired and must be stored and transmitted.

To address the limitations of conventional spectral imaging techniques such as spectrometers, many spectral imager sampling schemes based on compressive sensing [7, 20, 31] have been proposed [6, 41, 122]. The coded aperture snapshot spectral imager (CASSI) [3, 41, 115, 116] is a popular compressive spectral imager and acquires image data from different wavelengths simultaneously. In CASSI, the voxels of a scene are first coded by an aperture, then dispersed by a dispersive element, and finally detected by a 2D FPA. That is, a 3D image cube is suppressed and measured by a 2D array, and thus CASSI acquires far fewer measurements than those acquired by conventional spectral imagers, which significantly accelerates the imaging process. On the other hand, because the measurements from CASSI are highly compressive, reconstructing 3D image cubes from CASSI measurements becomes challenging. Moreover, because of the massive size of 3D image data, it is desirable to develop fast reconstruction algorithms in order to realize real time acquisition and processing.

Fortunately, it is possible to reconstruct the 3D cube from the 2D measurements according to the theory of compressive sensing [7, 20, 31], because the 2D images from different wavelengths are highly correlated, and the 3D image cube is sparse in an appropriate transform domain, meaning that only a small portion of the transform coefficients have large values. Therefore, we are motivated to investigate how to apply AMP to the CASSI system.

6.1 Related Work

Several algorithms have been proposed to reconstruct image cubes from measurements acquired by CASSI. One of the efficient algorithms is gradient projection for sparse reconstruction (GPSR) [38], which is fast and usually produces reasonable reconstruction. GPSR models hyperspectral image cubes as sparse in the Kronecker product of a 2D wavelet transform and a one-dimensional (1D) discrete cosine transform (DCT), and solves the regularized ℓ_1-minimization problem to enforce sparsity in this transform domain. Besides using ℓ_1-norm as the regularizer, total variation is a popular alternative; Wagadarikar et al. [116] employed total variation [21, 22] as the regularizer in the two-step iterative shrinkage/thresholding (TwIST) framework [14], a modified and fast version of standard iterative shrinkage/thresholding. Apart from using the wavelet-DCT basis, one can learn a dictionary with which the image cubes can be sparsely represented [74, 122]. An interesting idea to improve the reconstruction quality of the dictionary learning based approach is to use a standard image with red, green, and blue (RGB) components of the same scene as side information [122]. That is, a coupled dictionary is learned from the joint datasets of the CASSI measurements and the corresponding RGB image.

Despite the good results attained with the algorithms mentioned above, they all need manual tuning of some parameters, which may be time consuming. In GPSR and TwIST, the optimal regularization parameter could be different in reconstructing different image cubes; in dictionary learning methods, the patch size and the number of dictionary atoms must be chosen carefully.

6.2 Coded Aperture Snapshot Spectral Imager (CASSI)

6.2.1 Mathematical representation of CASSI

The coded aperture snapshot spectral imager (CASSI) [115] is a compressive spectral imaging system that collects far fewer measurements than traditional spectrometers. In CASSI, (*i*) the 2D spatial information of a scene is coded by an aperture, (*ii*) the coded spatial projections are spectrally shifted by a dispersive element, and (*iii*) the coded and shifted projections are detected by a 2D FPA. That is, in each coordinate of the FPA, the received projection is an integration of the coded and shifted voxels over all spectral bands at the same spatial coordinate. More specifically, let $f_0(a, b, \lambda)$ denote the density of a scene at spatial coordinate (a, b) and at wavelength λ, and let $T(a, b)$ denote the coded aperture. The coded density $T(a, b)f_0(a, b, \lambda)$ is then spectrally shifted by the dispersive element along one of the spatial dimensions. The energy received by the FPA at coordinate (a, b) is therefore

$$g(a,b) = \int_\Lambda T(a, b - S(\lambda))f_0(a, b - S(\lambda), \lambda)d\lambda, \tag{6.1}$$

where $S(\lambda)$ is the dispersion function induced by the prism at wavelength λ. Suppose we take a scene of spatial dimension m by n and spectral dimension l, i.e., the dimension of the image cube is $m \times n \times l$, and the dispersion is along the second spatial dimension y, then the number of measurements captured by the FPA will be $m(n + l - 1)$. If we approximate the integral in (6.1) by a discrete summation and vectorize the 3D image cube and the 2D measurements, then we obtain a matrix-vector form of (6.1),

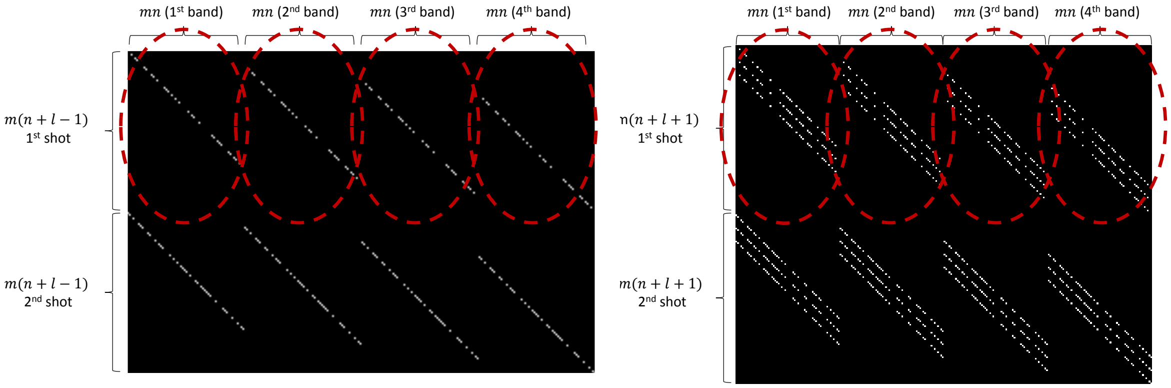
$$\mathbf{g} = \mathbf{H}\mathbf{f_0} + \mathbf{z}, \tag{6.2}$$

where $\mathbf{f_0}$ is the vectorized 3D image cube of dimension $N = mnl$, vectors $\mathbf{g}$ and $\mathbf{z}$ are the measurements and the additive white Gaussian noise, respectively, and the matrix $\mathbf{H}$ is an equivalent linear operator that models the integral in (6.1). A sketch of this matrix is depicted in Figure 6.1a when $K = 2$ shots are used.

6.2.2 Higher order CASSI

Recently, Arguello et al. [5] proposed a higher order model to characterize the CASSI system with greater precision, and improved the quality of the reconstructed 3D image cubes. In the standard CASSI system model, each cubic voxel in the 3D cube contributes to exactly one measurement in the FPA. In the higher order CASSI model, however, each cubic voxel is shifted to an oblique voxel because of the continuous nature of the dispersion, and therefore the oblique voxel contributes to more than one measurement in the FPA. As a result, the matrix $\mathbf{H}$ in (6.2) will have multiple diagonals as shown in Figure 6.1b, where there are sets of 3 diagonals for each FPA shot, accounting for the voxel energy impinging into the neighboring FPA pixels. In this case, the number of measurements with $K = 1$ shot of CASSI will be $M = m(n+l+1)$, because each diagonal entails the use of m more pixels (we refer readers to [5] for details).

In Section 6.4, we will provide promising image reconstruction results for this higher order CASSI system. Using the standard CASSI model, our proposed algorithm produces similar



(a) The matrix **H** for standard CASSI

(b) The matrix **H** for higher order CASSI

Figure 6.1: *The matrix* $\mathbf{H}$ *is presented for* $K = 2, m = n = 8$, *and* $l = 4$. *The circled diagonal patterns that repeat horizontally correspond to the coded aperture pattern used in the first FPA shot. The second coded aperture pattern determines the next set of diagonals. In (a) standard CASSI, each FPA shot captures* $m(n+l-1) = 88$ *measurements; in (b) higher order CASSI, each FPA shot captures* $m(n+l+1) = 104$ *measurements.*

advantageous results over other competing algorithms.

6.3 Proposed Algorithm

The goal of our proposed algorithm is to reconstruct the image cube $\mathbf{f_0}$ from its compressive measurements $\mathbf{g}$, where the matrix $\mathbf{H}$ is known. In this section, we describe our algorithm in detail. The algorithm employs (*i*) approximate message passing (AMP) [33], an iterative algorithm for compressive sensing problems, and (*ii*) a modified version of adaptive Wiener filtering, a hyperspectral image denoiser that can be applied within each iteration of AMP.

The CASSI system in (6.2) is a specific case of the system in (1.1), and the AMP framework described in (2.16) and (2.17) in Section 2.2.2 can be applied to the reconstruction problem for CASSI systems. In CASSI systems, The AMP algorithm proceeds as follows

$$\mathbf{f}^{t+1} = \eta_t(\mathbf{H}^T\mathbf{r}^t + \mathbf{f}^t), \tag{6.3}$$

$$\mathbf{r}^t = \mathbf{g} - \mathbf{H}\mathbf{f}^t + \frac{1}{R}\mathbf{r}^{t-1}\langle \eta_{t-1}'(\mathbf{H}^T\mathbf{r}^{t-1} + \mathbf{f}^{t-1})\rangle. \tag{6.4}$$

At iteration t, the function $\eta_t(\cdot)$ is a 3D image denoiser that is applied to a scalar channel,

$$\mathbf{q}^t = \mathbf{H}^T\mathbf{r}^t + \mathbf{f}^t = \mathbf{f_0} + \mathbf{v}^t, \tag{6.5}$$

where the variance of the noise $\mathbf{v}^t$ is estimated by (5.3) [66].

6.3.1 Adaptive Wiener filter

We are now ready to describe our 3D image denoiser, which is the function $\eta_t(\cdot)$ in the first step of AMP iterations in (6.3). Recall that in 2D image denoising problems, a 2D wavelet transform is often performed and some shrinkage function is applied to the wavelet coefficients in order to suppress noise [32, 37]. The wavelet transform based image denoising method is effective, because natural images are usually sparse in the wavelet transform domain, i.e., there are only a few large wavelet coefficients and the rest of the coefficients are small. Therefore, large wavelet coefficients are likely to contain information about the image, whereas small coefficients are usually comprised mostly of noise, and so it is effective to denoise by shrinking the small coefficients toward zero and suppressing the large coefficients according to the noise variance. Similarly, in hyperspectral image denoising, we want to find a sparsifying transform such that hyperspectral images have only a few large coefficients in this transform domain. Inspired by Arguello and Arce [4], we apply a wavelet transform to each of the 2D images in a 3D cube, and then apply a discrete cosine transform (DCT) along the spectral dimension, because the 2D slices from different wavelengths are highly correlated. That is, the sparsifying transform $\mathbf{\Psi}$ can be expressed as a Kronecker product of a DCT transform $\mathbf{\Phi}$ and a 2D wavelet transform $\mathbf{W}$, i.e., $\mathbf{\Psi} = \mathbf{\Phi} \otimes \mathbf{W}$, and it can easily be shown that $\mathbf{\Psi}$ is an orthonormal transform. Our 3D image denoising is processed after applying the sparsifying transform $\mathbf{\Psi}$ to the noisy image cube $\mathbf{q}^t$.

In Chapter 5, one of the image denoisers we employed was an adaptive Wiener filter in the wavelet domain, where the variance of each wavelet coefficient was estimated from its neighboring coefficients within a 5×5 window, i.e., the variance was estimated locally. Such an image denoiser performed well in the 2D compressive imaging problem, because the scalar channel (6.5) obtained from the AMP iterations (6.3,6.4) was an additive white Gaussian noise channel, and each wavelet coefficient contained independent and identically distributed (i.i.d.) Gaussian noise. In the CASSI system (6.2), however, because the matrix $\mathbf{H}$ is ill-conditioned as shown in Figure 6.1, the scalar channel (6.5) that is produced by AMP iterations is not additive white Gaussian, and the noisy 3D image cube $\mathbf{q}^t$ contains non-i.i.d. Gaussian noise. Consequently, estimating the coefficient variance from its small neighboring coefficients (a 3×3 or 5×5 neighboring window) may not be accurate. Therefore, we modify the local variance estimation to a global estimation procedure within each wavelet subband, and our simulation results show that global estimation provides better reconstruction quality than local estimation for hyperspectral images. Specifically, let $\theta_{\mathbf{q}}^t$ denote the coefficients of $\mathbf{q}^t$ in the transform domain, i.e., $\theta_{\mathbf{q}}^t = \mathbf{\Psi}\mathbf{q}$, and $\theta_{\mathbf{q},i}^t$ is the i-th element of $\theta_{\mathbf{q}}^t$. The coefficients $\widehat{\theta}_{\mathbf{f}}^t$ of the estimated (denoised) image cube $\mathbf{f}^t$ are obtained by Wiener filtering,

$$\widehat{\theta}^t_{\mathbf{f},i} = \frac{\max\{0, \widehat{\nu}^2_{i,t} - \widehat{\sigma}^2_t\}}{(\widehat{\nu}^2_{i,t} - \widehat{\sigma}^2_t) + \widehat{\sigma}^2_t} \left(\theta^t_{\mathbf{q},i} - \widehat{\mu}_{i,t}\right) + \widehat{\mu}_{i,t} = \frac{\max\{0, \widehat{\nu}^2_{i,t} - \widehat{\sigma}^2_t\}}{\widehat{\nu}^2_{i,t}} \left(\theta^t_{\mathbf{q},i} - \widehat{\mu}_{i,t}\right) + \widehat{\mu}_{i,t}, \qquad (6.6)$$

where $\widehat{\mu}_{i,t}$ and $\widehat{\nu}^2_{i,t}$ are the empirical mean and variance of $\theta^t_{\mathbf{q},i}$ within an appropriate wavelet subband, respectively. Taking the maximum between 0 and $(\widehat{\nu}^2_{i,t} - \widehat{\sigma}^2_t)$ ensures that if the empirical variance $\widehat{\nu}^2_{i,t}$ of the noisy coefficients is smaller than the noise variance $\widehat{\sigma}^2_t$, then the corresponding noisy coefficients are set to 0. After obtaining the denoised coefficients $\widehat{\theta}^t_{\mathbf{f}}$, the estimated image cube in the t-th iteration satisfies $\mathbf{f}^t = \mathbf{\Psi}^{-1}\widehat{\theta}^t_{\mathbf{f}} = \mathbf{\Psi}^T\widehat{\theta}^t_{\mathbf{f}}$.

6.3.2 Derivative of adaptive Wiener filter

The adaptive Wiener filter described in Section 6.3.1 is applied in (6.3) as the 3D image denoising function $\eta_t(\cdot)$. The following step in (6.4) requires $\eta_t'(\cdot)$, i.e., the derivative of $\eta_t(\cdot)$. We now show how to obtain $\eta_t'(\cdot)$. It has been discussed in Chapter 5 (see also Tan et al. [102]) that when the sparsifying transform is orthonormal, the derivative calculated in the transform domain is equivalent to the derivative in the image domain. According to (6.6), the derivative of the Wiener filter in the transform domain with respect to $\widehat{\theta}^t_{\mathbf{q},i}$ is $\max\{0, \widehat{\nu}^2_{i,t} - \widehat{\sigma}^2_t\}/\widehat{\nu}^2_{i,t}$. Because the sparsifying transform $\mathbf{\Psi}$ is orthonormal, the Onsager term in (6.4) can be calculated efficiently as

$$\langle \eta_t'(\mathbf{H}^T\mathbf{r}^t + \mathbf{f}^t)\rangle = \frac{1}{n}\sum_{i\in\mathcal{I}} \frac{\max\{0, \widehat{\nu}^2_{i,t} - \widehat{\sigma}^2_t\}}{\widehat{\nu}^2_{i,t}}, \qquad (6.7)$$

where $\mathcal{I}$ is the index set of all image cube elements, and the cardinality of $\mathcal{I}$ is $N = mnl$.

We focus on image denoising in an orthonormal transform domain and apply Wiener filtering to suppress noise, because it is convenient to obtain the Onsager correction term in (6.4). On the other hand, other denoisers that are not wavelet-DCT based can also be applied within the AMP framework. Metzler et al. [65], for example, proposed to utilize a block matching and 3D filtering denoising scheme (BM3D) [29] within AMP for 2D compressive imaging reconstruction, and run Monte Carlo [75] to approximate the Onsager correction term. However, the Monte Carlo technique is accurate only when the scalar channel (6.5) is Gaussian. In the CASSI system model (6.2), BM4D [62] may be an option for the 3D image denoising procedure. However, because the matrix $\mathbf{H}$ is ill-conditioned, the scalar channel (6.5) that is produced by AMP iterations (6.3,6.4) is not Gaussian, and thus the Monte Carlo technique fails to approximate the Onsager correction term.

6.3.3 AMP-3D-Wiener

The matrix $\mathbf{H}$ defined in (6.2) is not i.i.d. Gaussian, but highly structured as shown in Figure 6.1. The AMP framework may encounter divergence issues with this matrix $\mathbf{H}$. We introduced the standard technique "damping" [84, 114] in Section 5.2.2 to solve for the divergence problems of AMP, because it is simple and only increases the runtime modestly. AMP has been proved [84] to converge with sufficient damping, under the assumption that the prior of $\mathbf{f_0}$ has fixed means and variances throughout all iterations, and the amount of damping depends on the condition number of the matrix $\mathbf{H}$. Unfortunately, our previously proposed AMP-Wiener in Chapter 5 encounters divergence issues for the matrix $\mathbf{H}$ in (6.2) even with significant damping. Therefore, in our modified algorithm AMP-3D-Wiener, we propose a simple version of the adaptive Wiener filter as described in Section 6.3.1 to stabilize the estimation of the prior distribution of $\mathbf{f_0}$. We find in our simulations that AMP-3D-Wiener converges for all tested hyperspectral image cubes with a moderate amount of damping.

Our proposed AMP-3D-Wiener is summarized in Algorithm 2, where $\widehat{\mathbf{f}}_{\text{AMP}}$ denotes the image cube reconstructed by AMP-3D-Wiener. Note that in the first iteration of Algorithm 2, initialization of $\mathbf{q}^0$ and $\widehat{\nu}_0^2$ may not be necessary, because $\mathbf{r}^0$ is an all-zero vector, and the Onsager term is 0 at iteration 1.

```
**Algorithm 2** AMP-3D-Wiener
**Inputs:** $\mathbf{g}$, $\mathbf{H}$, $\alpha$, maxIter
**Outputs:** $\widehat{\mathbf{f}}_{\text{AMP}}$
**Initialization:** $\mathbf{f}^1 = \mathbf{0}$, $\mathbf{r}^0 = \mathbf{0}$
  **for** $t = 1 : \text{maxIter}$ **do**
    1. $\mathbf{r}^t = \mathbf{g} - \mathbf{H}\mathbf{f}^t + \frac{1}{R}\mathbf{r}^{t-1}\frac{1}{n}\sum_{i=1}^{n}\frac{\max\{0,\widehat{\nu}_{i,t-1}^2-\widehat{\sigma}_{t-1}^2\}}{\widehat{\nu}_{i,t-1}^2}$    % Residual
    2. $\mathbf{r}^t = \alpha \cdot \mathbf{r}^t + (1-\alpha) \cdot \mathbf{r}^{t-1}$    % Damping
    3. $\mathbf{q}^t = \mathbf{H}^T\mathbf{r}^t + \mathbf{f}^t$    % Noisy 3D image cube
    4. $\widehat{\nu}_t^2 = \frac{1}{m}\sum_j (r_j^t)^2$    % Noise variance
    5. $\theta_{\mathbf{q}}^t = \mathbf{\Psi}\mathbf{q}^t$    % Sparsifying transform
    6. $\widehat{\theta}_{\mathbf{f},i}^t = \frac{\max\{0,\widehat{\nu}_{i,t}^2-\widehat{\sigma}_t^2\}}{\widehat{\nu}_{i,t}^2}\left(\theta_{\mathbf{q},i}^t - \widehat{\mu}_{i,t}\right) + \widehat{\mu}_{i,t}$    % Adaptive Wiener filter
    7. $\mathbf{f}^{t+1} = \alpha \cdot \mathbf{\Psi}^T\widehat{\theta}_{\mathbf{f}}^t + (1-\alpha) \cdot \mathbf{f}^t$    % Damping
  **end for**
  $\widehat{\mathbf{f}}_{\text{AMP}} = \mathbf{f}^{\text{maxIter}+1}$
```

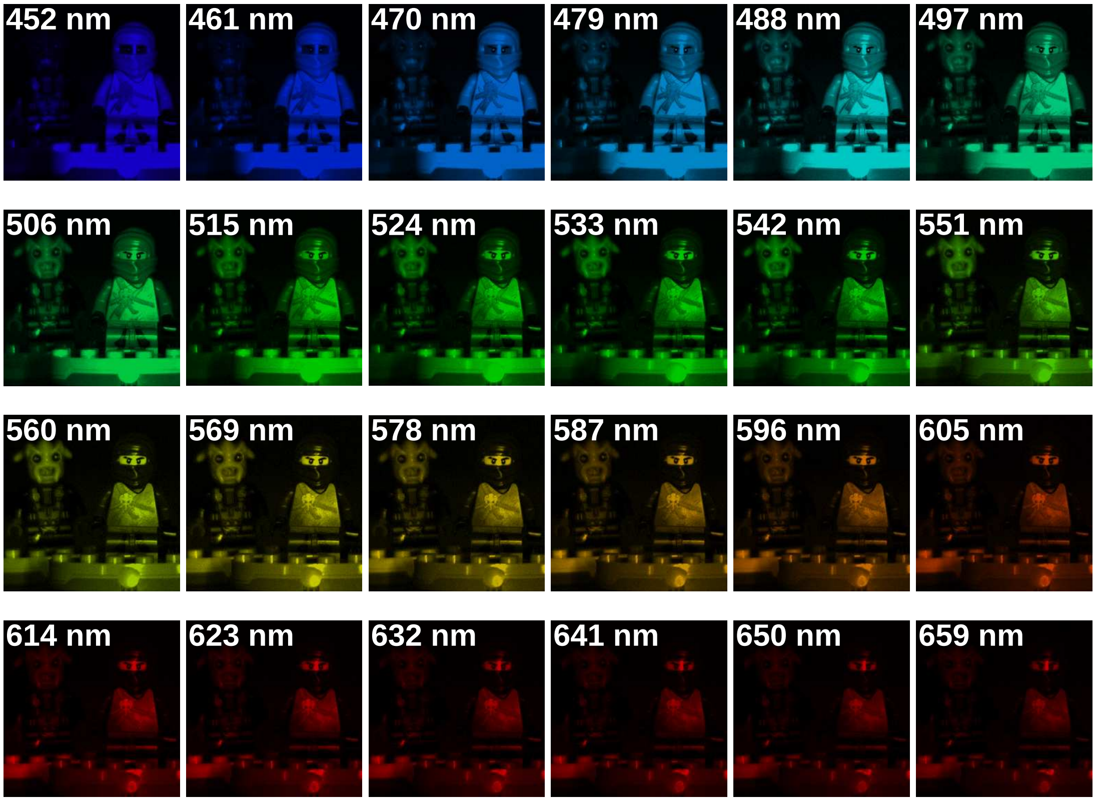


Figure 6.2: *The Lego scene. (The target object presented in the experimental results was not endorsed by the trademark owners and it is used here as fair use to illustrate the quality of reconstruction of compressive spectral image measurements. LEGO is a trademark of the LEGO Group, which does not sponsor, authorize or endorse the images in this dissertation. The LEGO Group. All Rights Reserved. http://aboutus.lego.com/en-us/legal-notice/fair-play/.)*

6.4 Numerical Results

In this section, we provide numerical results where we compare the reconstruction quality and runtime of AMP-3D-Wiener, gradient projection for sparse reconstruction (GPSR) [38], and two-step iterative shrinkage/thresholding (TwIST) [14, 116]. In all experiments, we use the same coded aperture pattern for AMP-3D-Wiener, GPSR, and TwIST. In order to quantify the reconstruction quality of each algorithm, the peak signal to noise ratio (PSNR) of each 2D slice in the reconstructed cubes is measured. The PSNR is defined as the ratio between the maximum squared value of the ground truth image cube $\mathbf{f_0}$ and the mean square error of the estimation $\widehat{\mathbf{f}}$, i.e.,

$$\text{PSNR} = 10 \cdot \log_{10} \left(\frac{\max_{x,y,\lambda} \left(f^2_{0,(x,y,\lambda)} \right)}{\sum_{x,y,\lambda} \left(\widehat{f}_{(x,y,\lambda)} - f_{0,(x,y,\lambda)} \right)^2} \right),$$

where $f_{(x,y,\lambda)}$ denotes the element in the cube $\mathbf{f}$ at spatial coordinate (x, y) and spectral coordinate λ.

In AMP, the damping parameter α is set to be 0.2. Other damping values are possible, but for smaller values such as 0.1, the reconstruction quality improves more slowly as the iteration number increases; for larger values such as 0.5, AMP may diverge. Recall that the amount of damping can be adjusted by evaluating the values of $\widehat{\nu}^2_t$ from (5.3). The choice of damping mainly depends on the structure of the imaging model in (6.2) but not on the characteristics of the image cubes, and thus the value of the damping parameter α need not be tuned in our experiments.

To reconstruct the image cube $\mathbf{f_0}$, GPSR and TwIST minimize objective functions of the form

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \frac{1}{2} \|\mathbf{g} - \mathbf{Hf}\|^2_2 + \beta \cdot \phi(\mathbf{f}), \tag{6.8}$$

where $\phi(\cdot)$ is a regularization function that characterizes the structure of the image cube $\mathbf{f_0}$, and β is a regularization parameter that balances the weights of the two terms in the objective function. In GPSR, $\phi(\mathbf{f}) = \|\mathbf{\Psi} f\|_1$; in TwIST, the total variation regularizer is employed,

$$\phi(\mathbf{f}) = \sum_{\lambda=1}^{L} \sum_{x=1}^{m} \sum_{y=1}^{n} \left((f(x+1, y, \lambda) - f(x, y, \lambda))^2 + (f(x, y+1, \lambda) - f(x, y, \lambda))^2 \right)^{1/2}. \tag{6.9}$$

The value of the regularization parameter β in (6.8) greatly affects the reconstruction results of GPSR and TwIST, and must be tuned carefully. We select the optimal values of β for GPSR and TwIST manually, i.e., we run GPSR and TwIST with $5-10$ different values of β, and select the results with the highest PSNR. The typical value of the regularization parameter for GPSR is between 10^{-5} and 10^{-4}, and the value for TwIST is around 0.1. We note in passing that the ground truth image cube is not known in practice, and estimating the PSNR obtained using different β may be quite involved and require oracle-like information when using GPSR and TwIST. There exist other hyperspectral image reconstruction algorithms based on dictionary learning [74, 122]. However, it is not straightforward to modify these dictionary learning methods to the higher order CASSI model described in Section 6.2.2, and so we do not include these algorithms in the comparison.

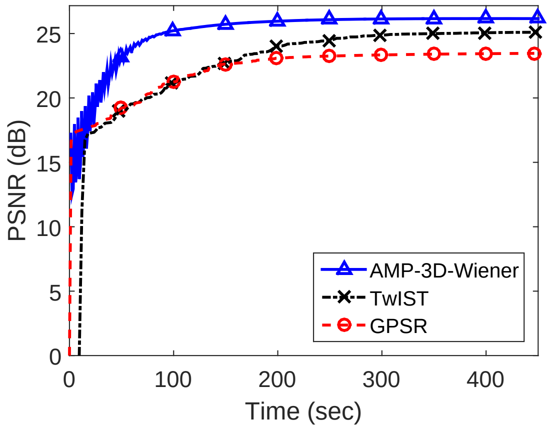


Figure 6.3: *Runtime versus average PSNR comparison of AMP-3D-Wiener, GPSR, and TwIST for the Lego image cube. Cube size is $m = n = 256$, and $l = 24$. The measurements are captured with $K = 2$ shots using complementary coded apertures, and the number of measurements is $M = 143,872$. Random Gaussian noise is added to the measurements such that the SNR is 20 dB.*

6.4.1 Test on "Lego" image cube

The first set of simulations is performed for the scene shown in Figure 6.2. This data cube was acquired using a wide-band Xenon lamp as the illumination source, modulated by a visible monochromator spanning the spectral range between 448 nm and 664 nm, and each spectral band has 9 nm width. The image intensity was captured using a grayscale CCD camera, with pixel size 9.9 μm, and 8 bits of intensity levels. The resulting test data cube has $m \times n = 256 \times 256$ pixels of spatial resolution and $l = 24$ spectral bands.

Setting 1: The measurements $\mathbf{g}$ are captured with $K = 2$ shots such that the coded aperture in the second shot is the complement of the aperture in the first shot. Therefore, we ensure that in the matrix $\mathbf{H}$ in (6.2), the norm of each column is similar. The measurement rate with two shots is $M/N = Km(n + l + 1)/(mnl) \approx 0.09$. Moreover, we add Gaussian noise with zero mean to the measurements. The signal to noise ratio (SNR) is defined as $10 \log_{10}(\mu_g/\sigma_{\text{noise}})$ [4], where μ_g is the mean value of the measurements $\mathbf{Hf_0}$ and σ_{noise} is the standard deviation of the additive noise $\mathbf{z}$. In Setting 1, we add measurement noise such that the SNR is 20 dB.

Figure 6.3 compares the reconstruction quality of AMP-3D-Wiener, GPSR, and TwIST within a certain amount of runtime. Runtime is measured on a Dell OPTIPLEX 9010 running an Intel(R) CoreTM i7-860 with 16GB RAM, and the environment is Matlab R2013a. In Figure 6.3, the horizontal axis represents runtime in seconds, and the vertical axis is the averaged PSNR over the 24 spectral bands. Although the PSNR of AMP-3D-Wiener oscillates at the first few iterations, which may be because the matrix $\mathbf{H}$ is ill-conditioned, it becomes stable after 50 seconds and reaches a higher level when compared to the PSNRs of GPSR and TwIST at 50

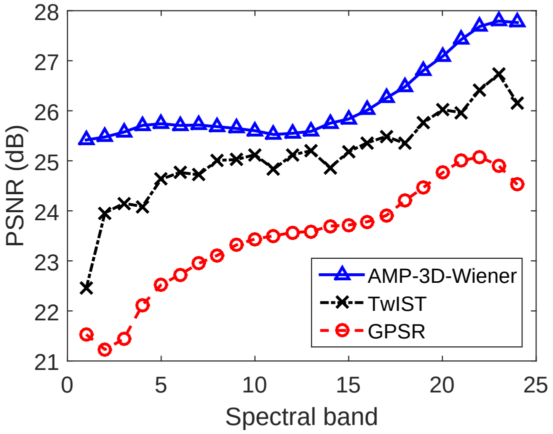


Figure 6.4: *Spectral band versus PSNR comparison of AMP-3D-Wiener, GPSR, and TwIST for the Lego image cube. Cube size is $m = n = 256$, and $l = 24$. The measurements are captured with $K = 2$ shots using complementary coded apertures, and the number of measurements is $M = 143,872$. Random Gaussian noise is added to the measurements such that the SNR is 20 dB.*

seconds. After 450 seconds, the average PSNR of the cube reconstructed by AMP-3D-Wiener (solid curve with triangle markers) is 26.16 dB, while the average PSNRs of GPSR (dash curve with circle markers) and TwIST (dash-dotted curve with cross markers) are 23.46 dB and 25.10 dB, respectively. Note that in 450 seconds, TwIST runs around 200 iterations, while AMP-3D-Wiener and GPSR run 400 iterations.

Figure 6.4 complements Figure 6.3 by illustrating the PSNR of each 2D slice in the reconstructed cube separately. It is shown that the cube reconstructed by AMP-3D-Wiener has $2-4$ dB higher PSNR than the cubes reconstructed by GPSR and $0.4-3$ dB higher than those of TwIST for all 24 slices.

In Figure 6.5, we plot the 2D slices at wavelengths 488 nm, 533 nm, and 578 nm in the actual image cubes reconstructed by AMP-3D-Wiener, GPSR, and TwIST. The images in these four rows are slices from the ground truth image cube $\mathbf{f_0}$, the cubes reconstructed by AMP-3D-Wiener, GPSR, and TwIST, respectively. At the same time, the images in columns $1-3$ show the upper-left part of the scene, whereas images in columns $4-6$ show the upper-right part of the scene. All images are of size 128×128. It is clear from Figure 6.5 that the 2D slices reconstructed by AMP-3D-Wiener have better visual quality; the slices reconstructed by GPSR have blurry edges, and the slices reconstructed by TwIST lack details, because the total variation regularization tends to constrain the images to be piecewise constant.

Furthermore, a spectral signature plot analyzes how the pixel values change along the spectral dimension at a fixed spatial location, and we present such spectral signature plots for the image cubes reconstructed by AMP-3D-Wiener, GPSR, and TwIST in Figure 6.6. Three spatial

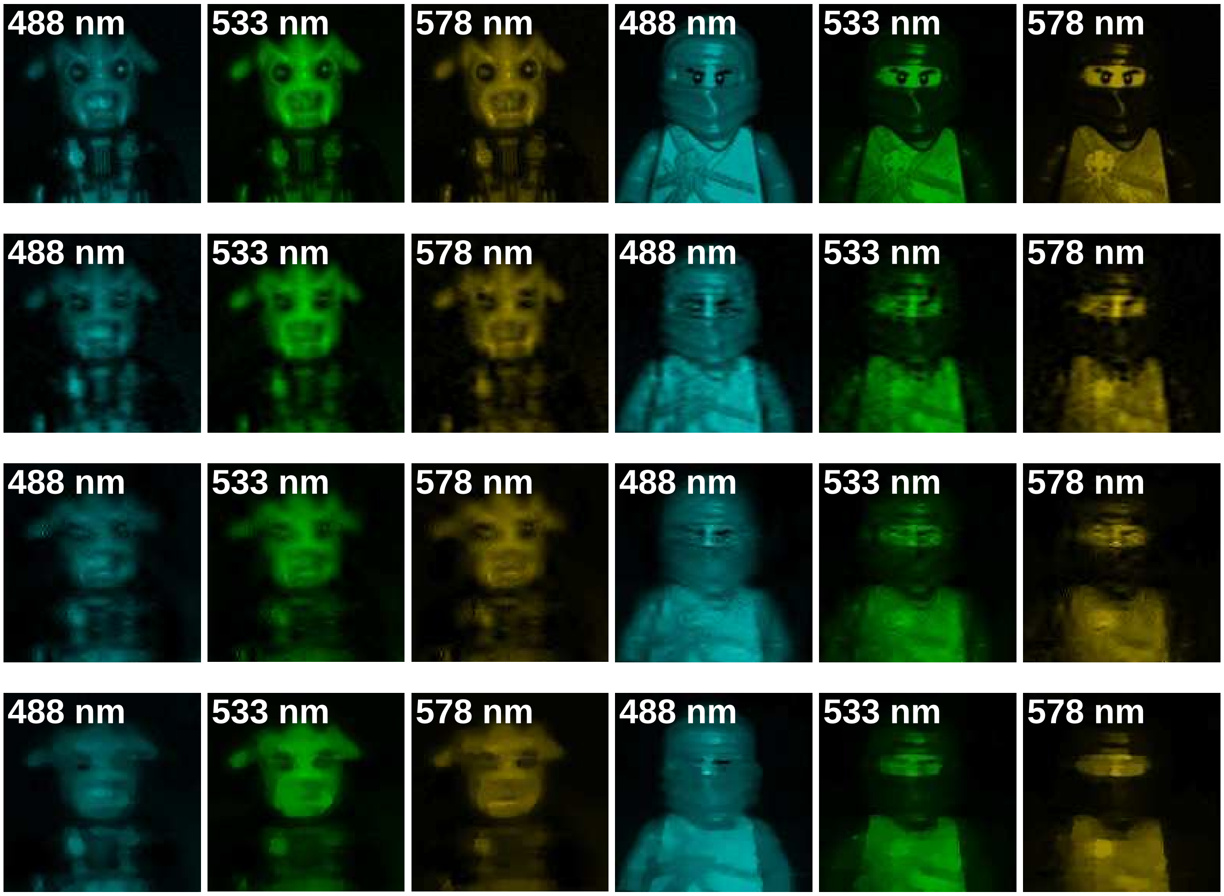
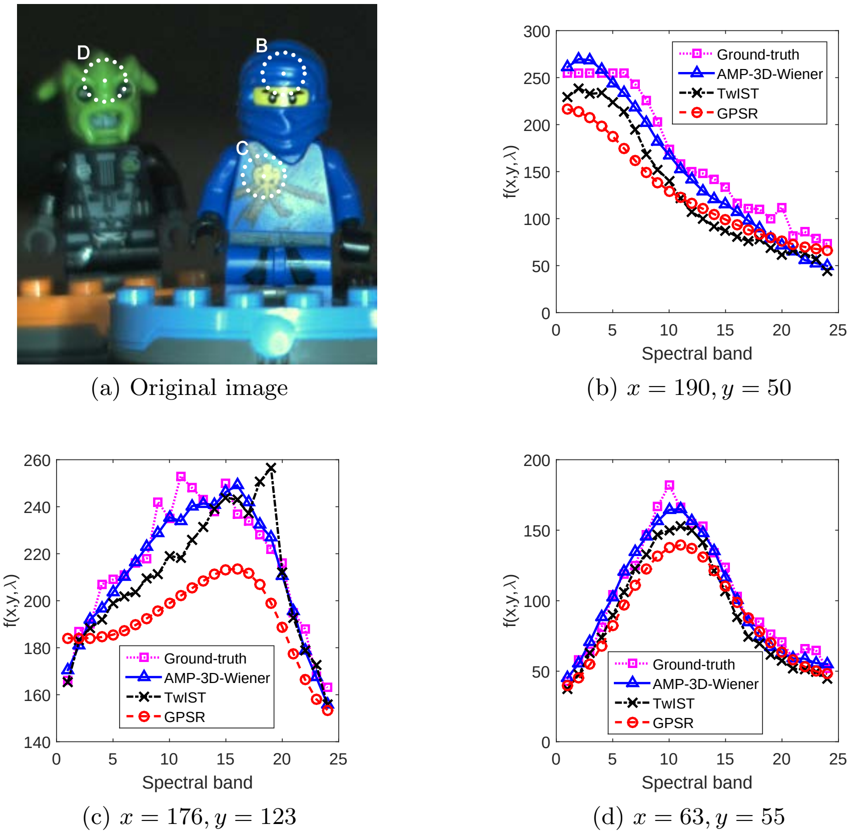


Figure 6.5: *2D slices at wavelengths* 488 *nm,* 533 *nm, and* 578 *nm in the image cubes reconstructed by AMP-3D-Wiener, GPSR, and TwIST for the Lego image cube. Cube size is* $m = n = 256$*, and* $l = 24$*. The measurements are captured with* $K = 2$ *shots using complementary coded apertures, and the number of measurements is* $M = 143,872$*. Random Gaussian noise is added to the measurements such that the SNR is 20 dB. First row: ground truth; second row: the reconstruction result by AMP-3D-Wiener; third row: the reconstruction result by GPSR; and last row: the reconstruction result by TwIST. Columns* $1-3$*: upper-left part of the scene of size* 128×128*; and columns* $4-6$*: upper-right part of the scene of size* 128×128*.*

locations are selected as shown in Figure 6.6a, and the spectral signature plots for locations B, C, and D are shown in Figures 6.6b–6.6d, respectively. It can be seen that the spectral signatures of the cube reconstructed by AMP-3D-Wiener closely resemble those of the ground truth image cube (dotted curve with square markers), whereas there are discrepancies between the spectral signatures of the cube reconstructed by GPSR or TwIST and those of the ground truth cube.



(a) Original image (b) $x = 190, y = 50$

(c) $x = 176, y = 123$ (d) $x = 63, y = 55$

Figure 6.6: *Comparison of AMP-3D-Wiener, GPSR, and TwIST on reconstruction along the spectral dimension of three spatial pixel locations as indicated in (a). The estimated pixel values are illustrated for (b) the pixel B, (c) the pixel C, and (d) the pixel D.*

According to the runtime experiment from Setting 1, we run AMP-3D-Wiener with 400 iterations, GPSR with 400 iterations, and TwIST with 200 iterations for the rest of the simulations, so that all algorithms complete within a similar amount of time.

Setting 2: In this experiment, we add measurement noise such that the SNR varies from 15 dB to 40 dB, which is the same setting as in Arguello and Arce [4], and the result is shown in Figure 6.7. Again, AMP-3D-Wiener achieves more than 2 dB higher PSNR than GPSR, and about 1 dB higher PSNR than TwIST, overall.

Setting 3: In Settings 1 and 2, the measurements are captured with $K = 2$ shots. We now test our algorithm on the setting where the number of shots varies from $K = 2$ to $K = 12$ with pairwise complementary coded apertures. Specifically, we randomly generate the coded aperture for the k-th shot for $k = 1, 3, 5, 7, 9, 11$, and the coded aperture in the $(k + 1)$-th shot is the complement of the aperture in the k-th shot. In this setting, a moderate amount of noise (20 dB) is added to the measurements. Figure 6.8 presents the PSNR of the reconstructed

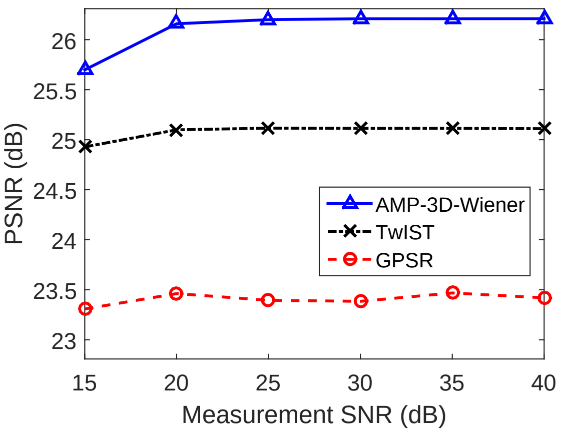


Figure 6.7: *Measurement noise versus average PSNR comparison of AMP-3D-Wiener, GPSR, and TwIST for the Lego image cube. Cube size is $m = n = 256$, and $l = 24$. The measurements are captured with $K = 2$ shots using complementary coded apertures, and the number of measurements is $M = 143,872$.*

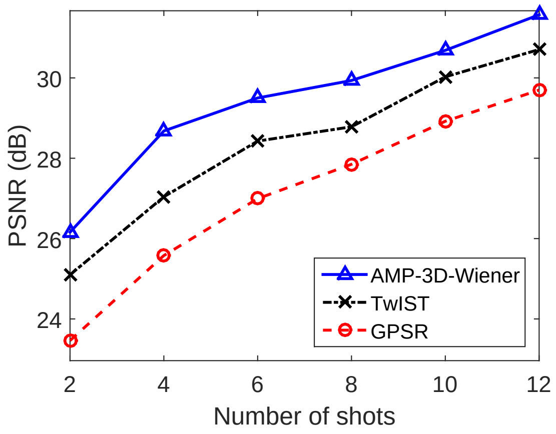


Figure 6.8: *Number of shots versus average PSNR comparison of AMP-3D-Wiener, GPSR, and TwIST for the Lego image cube. Cube size is $n = m = 256$, and $l = 24$. The measurements are captured using pairwise complementary coded apertures. Random Gaussian noise is added to the measurements such that the SNR is 20 dB.*

cubes as a function of the number of shots, and AMP-3D-Wiener consistently beats GPSR and TwIST.

	15 dB			20 dB		
	AMP	GPSR	TwIST	AMP	GPSR	TwIST
Scene 1	**32.69**	28.10	31.05	**33.29**	28.09	31.16
Scene 2	**26.52**	24.32	26.25	**26.65**	24.40	26.41
Scene 3	**32.05**	29.33	31.21	**32.45**	29.55	31.54
Scene 4	**27.57**	25.19	27.17	**27.76**	25.47	27.70
Scene 5	**29.68**	27.09	29.07	**29.80**	27.29	29.42
Scene 8	**28.72**	25.53	26.24	**29.33**	25.77	26.46

Table 6.1: *Average PSNR comparison of AMP-3D-Wiener, GPSR, and TwIST for the dataset "natural scene 2002" downloaded from [39]. The spatial dimensions of the cubes are cropped to* $m = n = 512$*, and each cube has* $l = 31$ *spectral bands. The measurements are captured with* $K = 2$ *shots, and the number of measurements is* $M = 557,056$*. Random Gaussian noise is added to the measurements such that the SNR is 15 or 20 dB. Because the spatial dimensions of the cubes "scene 6" and "scene7" in "natural scenes 2002" are smaller than* 512×512*, we do not include results for these two cubes.*

6.4.2 Test on natural scenes

Besides the Lego image cube, we have also tested our algorithm on image cubes of natural scenes [39].[1] There are two datasets, "natural scenes 2002" and "natural scenes 2004," each one with 8 image data cubes. The cubes in the first dataset have $L = 31$ spectral bands with spatial resolution of around 700×700, whereas the cubes in the second dataset have $L = 33$ spectral bands with spatial resolution of around 1000×1000. To satisfy the dyadic constraint of the 2D wavelet, we crop their spatial resolution to be $m = n = 512$. Because the spatial dimensions of the cubes "scene 6" and "scene7" in the first dataset are smaller than 512×512, we do not include results for these two cubes.

The measurements are captured with $K = 2$ shots, and the measurement rate is $M/N = Km(n+l+1)/(mnl) \approx 0.069$ for "natural scene 2002" and 0.065 for "natural scene 2004." We test for measurement noise levels such that the SNRs are 15 dB and 20 dB. The typical runtimes for AMP with 400 iterations, GPSR with 400 iterations, and TwIST with 200 iterations are approximately $2,800$ seconds. The average PSNR over all spectral bands for each reconstructed cube is shown in Tables 6.1 and 6.2. We highlight the highest PSNR among AMP-3D-Wiener, GPSR, and TwIST using bold fonts. It can be seen from Tables 6.1 and 6.2 that AMP-3D-Wiener usually outperforms GPSR by $2-5$ dB in terms of the PSNR, and outperforms TwIST by $0.2-4$ dB. In addition, the results of 6 selected image cubes are displayed in Figure 6.9 in the form of 2D RGB images.[2] The four rows of images correspond to ground truth, results by AMP-3D-Wiener, results by GPSR, and results by TwIST, respectively. We can see from Fig-

[1] The cubes are downloaded from http://personalpages.manchester.ac.uk/staff/d.h.foster/.

[2] We refer to the tutorial from http://personalpages.manchester.ac.uk/staff/david.foster/Tutorial_HSI2RGB/Tutorial_HSI2RGB.html and convert 3D image cubes to 2D RGB images.

	15 dB			20 dB		
	AMP	GPSR	TwIST	AMP	GPSR	TwIST
Scene 1	**30.48**	28.43	30.17	**30.37**	28.53	30.31
Scene 2	**27.34**	24.71	27.03	**27.81**	24.87	27.35
Scene 3	**33.13**	29.38	31.69	**33.12**	29.44	31.75
Scene 4	**32.07**	26.99	31.69	**32.14**	27.25	32.08
Scene 5	**27.44**	24.25	26.48	**27.83**	24.60	26.85
Scene 6	**29.15**	24.99	25.74	**30.00**	25.53	26.15
Scene 7	**36.35**	33.09	33.59	**37.11**	33.55	34.05
Scene 8	**32.12**	28.14	28.22	**32.93**	28.82	28.69

Table 6.2: *Average PSNR comparison of AMP-3D-Wiener, GPSR, and TwIST for the dataset "natural scene 2004" downloaded from [39]. The spatial dimensions of the cubes are cropped to $m = n = 512$, and each cube has $l = 33$ spectral bands. The measurements are captured with $K = 2$ shots, and the number of measurements is $M = 559,104$. Random Gaussian noise is added to the measurements such that the SNR is 15 or 20 dB.*

ure 6.9 that AMP-3D-Wiener produces images with better quality, while images reconstructed by GPSR and TwIST are blurrier.

6.5 Conclusion

In this chapter, we considered the compressive hyperspectral imaging reconstruction problem for the coded aperture snapshot spectral imager (CASSI) system. Considering that the CASSI system is a great improvement in terms of imaging quality and acquisition speed over conventional spectral imaging techniques, it is desirable to further improve CASSI by accelerating the 3D image cube reconstruction process. Our proposed AMP-3D-Wiener used an adaptive Wiener filter as a 3D image denoiser within the approximate message passing (AMP) [33] framework. AMP-3D-Wiener was faster than existing image cube reconstruction algorithms, and also achieved better reconstruction quality.

Future improvements: In our current AMP-3D-Wiener algorithm for compressive hyperspectral imaging reconstruction, we estimated the noise variance of the noisy image cube within each AMP iteration using (5.3). In order to denoise the noisy image cube in the sparsifying transform domain, we applied the estimated noise variance value to all wavelet subbands. The noise variance estimation and 3D image denoising method were effective, and helped produce promising reconstruction. However, both the noise variance estimation and the 3D image denoising method may be sub-optimal, because the noisy image cube within each AMP iteration does not contain i.i.d. Gaussian noise, and so the coefficients in the different wavelet subbands may contain different amounts of noise. Therefore, it is possible that the denoising part of

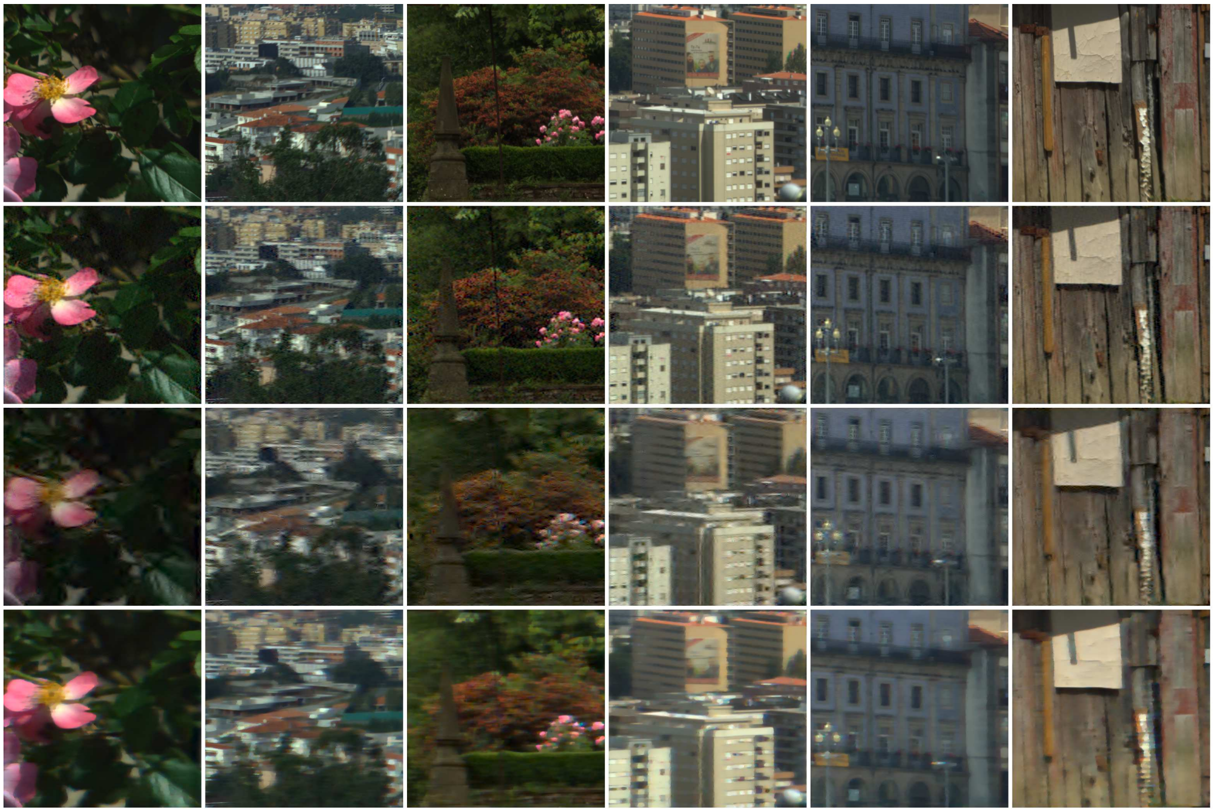


Figure 6.9: *Comparison of selected image cubes reconstructed by AMP-3D-Wiener, GPSR, and TwIST for the datasets "natural scene 2002" and "natural scene 2004." The 2D RGB images shown in this figure are converted from their corresponding 3D image cubes. Cube size is $n = m = 512$, and $l = 31$ for images in columns $1 - 2$ or $L = 33$ for images in columns $3 - 6$. Random Gaussian noise is added to the measurements such that the SNR is 20 dB. First row: ground truth; second row: the reconstruction result by AMP-3D-Wiener; third row: the reconstruction result by GPSR; and last row: the reconstruction result by TwIST.*

the proposed algorithm, including the noise variance estimation, can be further improved. The study of such denoising methods is left for future work.

Chapter 7

Discussion

In this dissertation, we explored several aspects of compressive signal reconstruction problems. First, we considered non-standard error metrics in order to robustify the reconstruction algorithms, because minimizing conventional error metrics such as mean square error may not always yield satisfactory results. Seeing that generalized approximate message passing (GAMP) and approximate message passing (AMP) decouple the compressive sensing system to parallel scalar Gaussian channels and provide the posterior distribution of the input signal of interest, we utilized this feature of GAMP and AMP to obtain the reconstructed signal by minimizing the expected user-defined error.

As an example of a non-standard error metric, we looked into the ℓ_∞-norm error. We found interesting properties of signal reconstruction by minimizing ℓ_∞-norm error, where input signals are generated by i.i.d. Gaussian mixture sources. More specifically, we found that in parallel scalar Gaussian channels, the minimum ℓ_∞-norm error is achieved by the Wiener filter that corresponds to the Gaussian component with the maximum variance. Since compressive sensing systems can be decoupled by GAMP and AMP, the Wiener filter can also be applied in compressive sensing systems to minimize the ℓ_∞-norm error. Additionally, we used ℓ_p-norm error as a practical alternative to ℓ_∞-norm error, owing to the fact that the theoretical results for ℓ_∞-norm error are asymptotic in nature. On the other hand, the theoretical results are based on the assumption that the input signals are generated by i.i.d. Gaussian mixture sources, and it would be interesting to investigate the properties of ℓ_∞-norm minimization where input signals are generated by other sources.

After looking into the compressive signal reconstruction problems with non-standard error metrics, we studied compressive imaging problems, which are important applications of compressive signal reconstruction. We developed AMP-based compressive image reconstruction algorithms, for both general compressive imaging settings and hyperspectral imaging settings. By employing robust and efficient image denoisers within AMP iterations, our proposed algorithms outperform the prior art in terms of both reconstruction quality and runtime.

One limitation of our proposed hyperspectral image reconstruction algorithm is that we restrict the coded apertures in multiple shots to be complementary. For completely random coded apertures, our proposed algorithm encounters divergence problems, and this problem is left for future work.

In our compressive image reconstruction algorithms, however, we used squared error (ℓ_2-norm error) as the performance metric. Besides the ℓ_2-norm error, Bayati and Montanari [10] have shown that state evolution in AMP also holds for other error metrics such as ℓ_p-norm error where $p \neq 2$. We believe that, when some error metric other than ℓ_2 error is considered, there exist denoisers that are optimal for this error metric of interest, and thus by applying these denoisers within AMP we may be able to achieve optimal reconstruction results for compressive image reconstruction problems. The development of such denoisers is a possible direction for future work.

REFERENCES

[1] M. Akçakaya and V. Tarokh. Shannon-theoretic limits on noisy compressive sampling. *IEEE Trans. Inf. Theory*, 56(1):492–504, Jan. 2010.

[2] T. Alecu, S. Voloshynovskiy, and T. Pun. The Gaussian transform of distributions: Definition, computation and application. *IEEE Trans. Signal Process.*, 54(8):2976–2985, Aug. 2006.

[3] H. Arguello and G. Arce. Code aperture optimization for spectrally agile compressive imaging. *J. Opt. Soc. Am.*, 28(11):2400–2413, Nov. 2011.

[4] H. Arguello and G. Arce. Colored coded aperture design by concentration of measure in compressive spectral imaging. *IEEE Trans. Image Process.*, 23(4):1896–1908, Mar. 2014.

[5] H. Arguello, H. Rueda, Y. Wu, D. Prather, and G. Arce. Higher-order computational model for coded aperture spectral imaging. *Appl. Optics*, 52(10):D12–D21, Mar. 2013.

[6] Y. August, C. Vachman, Y. Rivenson, and A. Stern. Compressive hyperspectral imaging by random separable projections in both the spatial and the spectral domains. *Appl. Optics*, 52(10):D46–D54, Mar. 2013.

[7] R. G. Baraniuk. A lecture on compressive sensing. *IEEE Signal Process. Mag.*, 24(4):118–121, July 2007.

[8] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Trans. Inf. Theory*, 56(4):1982–2001, Apr. 2010.

[9] D. Baron, S. Sarvotham, and R. G. Baraniuk. Bayesian compressive sensing via belief propagation. *IEEE Trans. Signal Process.*, 58:269–280, Jan. 2010.

[10] M. Bayati and A. Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Trans. Inf. Theory*, 57(2):764–785, Feb. 2011.

[11] R. Berinde, P. Indyk, and M. Ruzic. Practical near-optimal sparse recovery in the ℓ_1 norm. In *Proc. IEEE 46th Allerton Conf. Commun., Control, Comput.*, pages 198–205, Sept. 2008.

[12] S. Berman. A law of large numbers for the maximum in a stationary Gaussian sequence. *Ann. Math. Stat.*, 33(1):93–97, Mar. 1962.

[13] A. Bijaoui. Wavelets, Gaussian mixtures and Wiener filtering. *Signal Process.*, 82(4):709–712, Apr. 2002.

[14] J. Bioucas-Dias and M. Figueiredo. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Process.*, 16(12):2992–3004, Dec. 2007.

[15] C.M. Bishop. *Pattern recognition and machine learning*, volume 4. New York, NY, USA: Springer, 2006.

[16] R.E. Blahut. *Theory of remote image formation.* Cambridge University Press, 2004.

[17] M. Borgerding, P. Schniter, and S. Rangan. Generalized approximate message passing for cosparse analysis compressive sensing. *ArXiv preprint arXiv:1312.3968*, 2014.

[18] D. Brady. *Optical imaging and spectroscopy.* Hoboken, NJ: Wiley, 2009.

[19] G. Caire, R.R. Muller, and T. Tanaka. Iterative multiuser joint decoding: Optimal power allocation and low-complexity implementation. *Information Theory, IEEE Transactions on*, 50(9):1950–1973, Sept. 2004.

[20] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory*, 52(2):489–509, Feb. 2006.

[21] A. Chambolle. An algorithm for total variation minimization and applications. *J. Math. imaging vision*, 20(1-2):89–97, Jan. 2004.

[22] T. Chan, S. Esedoglu, F. Park, and A. Yip. Recent developments in total variation image restoration. *Math. Models Computer Vision*, 17, 2005.

[23] S. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Trans. Image Process.*, 9(9):1532–1546, Sept. 2000.

[24] S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comp.*, 20(1):33–61, 1998.

[25] J. F. Claerbout. *Imaging the earth's interior.* Blackwell Scientific Publications, 1985.

[26] C.E. Clark. The greatest of a finite set of random variables. *Oper. Res.*, 9(2):145–162, Mar. 1961.

[27] A. Cohen, W. Dahmen, and R. A. DeVore. Near optimal approximation of arbitrary vectors from highly incomplete measurements. 2007. preprint.

[28] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* New York, NY, USA: Wiley-Interscience, 2006.

[29] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.*, 16(8):2080–2095, Aug. 2007.

[30] M. Dalai and R. Leonardi. ℓ-infinity constrained approximations for image and video compression. In *Picture coding symp.*, Apr. 2006.

[31] D. Donoho. Compressed sensing. *IEEE Trans. Inf. Theory*, 52(4):1289–1306, Apr. 2006.

[32] D. L Donoho and J. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, Sept. 1994.

[33] D. L. Donoho, A. Maleki, and A. Montanari. Message passing algorithms for compressed sensing. *Proc. Nat. Academy Sci.*, 106(45):18914–18919, Nov. 2009.

[34] M. Egerstedt and C. F. Martin. Trajectory planning in the infinity norm for linear control systems. *Int. J. Control*, 72(13):1139–1146, 1999.

[35] M. Eismann. *Hyperspectral remote sensing.* Bellingham, WA: SPIE, 2012.

[36] L. Fang, S. Li, R McNabb, Qing Nie, A Kuo, C Toth, Joseph A Izatt, and Sina Farsiu. Fast acquisition and reconstruction of optical coherence tomography images via sparse representation. *IEEE Trans. Med. Imag.*, 32(11):2034–2049, Nov. 2013.

[37] M. Figueiredo and R. Nowak. Wavelet-based image estimation: An empirical Bayes approach using Jeffrey's noninformative prior. *IEEE Trans. Image Process.*, 10(9):1322–1331, Sept. 2001.

[38] M. Figueiredo, R. Nowak, and S. J. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J. Select. Topics Signal Proces.*, 1:586–597, Dec. 2007.

[39] D.H. Foster, K. Amano, S.M.C. Nascimento, and M.J. Foster. Frequency of metamerism in natural scenes. *J. Optical Soc. of Amer. A*, 23(10):2359–2372, Oct. 2006.

[40] N. Gat. Imaging spectroscopy using tunable filters: A review. In *Proc. SPIE*, volume 4056, pages 50–64, Apr. 2000.

[41] M. Gehm, R. John, D. Brady, R. Willett, and T. Schulz. Single-shot compressive spectral imaging with a dual-disperser architecture. *Opt. Exp.*, 15(21):14013–14027, Oct. 2007.

[42] A. C. Gilbert, B. Hemenway, A. Rudra, M. J. Strauss, and M. Wootters. Recovering simple signals. In *Inf. Theory Appl. Workshop*, pages 382–391, Feb. 2012.

[43] B. Gnedenko. Sur la distribution limite du terme maximum d'une série aléatoire. *Ann. Math.*, 44(3):423–453, July 1943.

[44] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm. *IEEE Trans. Signal Process.*, 45(3):600–616, Mar. 1997.

[45] U. Grenander and M. Rosenblatt. *Statistical analysis of stationary time series.* New York, NY, USA: Wiley, 1957.

[46] D. Guo, D. Baron, and S. Shamai. A single-letter characterization of optimal noisy compressed sensing. In *Proc. 47th Allerton Conf. Commun., Control, and Comput.*, pages 52–59, Sept. 2009.

[47] D. Guo and S. Verdú. Randomly spread CDMA: Asymptotics via statistical physics. *IEEE Trans. Inf. Theory*, 51(6):1983–2010, June 2005.

[48] D. Guo and C. Wang. Asymptotic mean-square optimality of belief propagation for sparse linear systems. In *IEEE Inf. Theory Workshop*, pages 194–198, Oct. 2006.

[49] D. Guo and C. C. Wang. Random sparse linear systems observed via arbitrary channels: A decoupling principle. In *Proc. IEEE Int. Symp. Inf. Theory*, pages 946–950, June 2007.

[50] D. Guo and C. C. Wang. Multiuser detection of sparsely spread CDMA. *IEEE J. Select. Areas Commun.*, 26(3):421–431, Apr. 2008.

[51] L. He and L. Carin. Exploiting structure in wavelet-based Bayesian compressive sensing. *IEEE Trans. Signal Process.*, 57(9):3488–3497, Sept. 2009.

[52] E Keith Hege, Dan O'Connell, William Johnson, Shridhar Basty, and Eustace L Dereniak. Hyperspectral imaging for astronomy and space surviellance. In *SPIE's 48th Annu. Meeting Opt. Sci. and Technol.*, pages 380–391, Jan. 2004.

[53] Daniel C Heinz and Chein-I Chang. Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.*, 39(3):529–545, Mar. 2001.

[54] P. Indyk. On approximate nearest neighbors under ℓ_∞ norm. *J. Comput. Syst. Sci.*, 63(4):627–638, Dec. 2001.

[55] P. Indyk and M. Ruzic. Near-optimal sparse recovery in the ℓ_1 norm. In *49th Annu. IEEE Symp. Found. Comput. Sci.*, pages 199–207, Oct. 2008.

[56] R. Jenatton, J. Mairal, F. Bach, and G. Obozinski. Proximal methods for sparse hierarchical dictionary learning. In *Proc. 27th Int. Conf. Mach. Learning*, pages 487–494, June 2010.

[57] X. Jia and J. Richards. Segmented principal components transformation for efficient hyperspectral remote-sensing image display and classification. *IEEE Trans. Geosci. Remote Sens.*, 37(1):538–542, Jan. 1999.

[58] M. K. Mıhçak, I. Kozintsev, K. Ramchandran, and P. Moulin. Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Process. Letters*, 6(12):300–303, Dec. 1999.

[59] F. Kruse, J. Boardman, and J. Huntington. Comparison of airborne hyperspectral data and EO-1 Hyperion for mineral mapping. *IEEE Trans. Geosci. Remote Sens.*, 41(6):1388–1400, June 2003.

[60] B.C. Levy. *Principles of signal detection and parameter estimation.* New York, NY, USA: Springer Verlag, 2008.

[61] Z. P. Liang and P. C. Lauterbur. *Principles of magnetic resonance imaging: a signal processing perspective.* SPIE Optical Engineering Press, 2000.

[62] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi. A nonlocal transform-domain filter for volumetric data denoising and reconstruction. *IEEE Trans Image Process.*, 22(1):119–133, Jan. 2013.

[63] S.G. Mallat. *A wavelet tour of signal processing.* Academic Press, 1999.

[64] A. Manoel, F. Krzakala, E. Tramel, and L. Zdeborová. Sparse estimation with the swept approximated message-passing algorithm. *Arxiv preprint arxiv:1406.4311*, June 2014.

[65] C. Metzler, A. Maleki, and R. G. Baraniuk. From denoising to compressed sensing. *Arxiv preprint arxiv:1406.4175v2*, June 2014.

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

[67] A. Montanari and D. Tse. Analysis of belief propagation for non-linear problems: The example of CDMA (or: How to prove Tanaka's formula). In *IEEE Inf. Theory Workshop*, pages 160–164, Mar. 2006.

[68] P. Moulin and J. Liu. Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors. *IEEE Trans. Inf. Theory*, 45(3):909–919, Apr. 1999.

[69] A. Mousavi, A. Maleki, and R. Baraniuk. Parameterless optimal approximate message passing. *Arxiv preprint arxiv:1311.0035*, Oct. 2013.

[70] F. Natterer and F. Wubbeling. *Mathematical methods in image reconstruction.* Society for Industrial Mathematics, 2001.

[71] D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Computational Harmonic Anal.*, 26(3):301–321, May 2009.

[72] S. Panasyuk, S. Yang, D. Faller, D. Ngo, R. Lew, J. Freeman, and A. Rogers. Medical hyperspectral imaging to facilitate residual tumor identification during surgery. *Cancer Biol. Therapy*, 6(3):439–446, Mar. 2007.

[73] A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw Hill Book Co., 1991.

[74] A. Rajwade, D. Kittle, T. Tsai, D. Brady, and L. Carin. Coded hyperspectral imaging and blind compressive sensing. *SIAM J. Imag. Sci.*, 6(2):782–812, Apr. 2013.

[75] S. Ramani, T. Blu, and M. Unser. Monte-Carlo SURE: A black-box optimization of regularization parameters for general denoising algorithms. *IEEE Trans. Image Process.*, 17(9):1540–1554, Sept. 2008.

[76] S. Rangan. Estimation with random linear mixing, belief propagation and compressed sensing. In *Proc. IEEE 44th Conference Inf. Sci. Syst. (CISS)*, Mar. 2010.

[77] S. Rangan. Estimation with random linear mixing, belief propagation and compressed sensing. *ArXiv preprint arXiv:1001.2228*, Jan. 2010.

[78] S. Rangan. Generalized approximate message passing for estimation with random linear mixing. *Arxiv preprint arXiv:1010.5141*, Oct. 2010.

[79] S. Rangan. Generalized approximate message passing for estimation with random linear mixing. In *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pages 2168–2172, July 2011.

[80] S. Rangan, A. Fletcher, V. Goyal, U. Kamilov, J. Parker, and P. Schniter. GAMP. `http://gampmatlab.wikia.com/wiki/Generalized_Approximate_Message_Passing/`.

[81] S. Rangan, A. K. Fletcher, and V. K. Goyal. Asymptotic analysis of MAP estimation via the replica method and applications to compressed sensing. *IEEE Trans. Inf. Theory*, 58(3):1902–1923, Mar. 2012.

[82] S. Rangan, A. K. Fletcher, P. Schniter, and U. Kamilov. Inference for generalized linear models via alternating directions and Bethe free energy minimization. *Arxiv preprint arxiv:1501.01797*, Jan. 2015.

[83] S. Rangan, A.K. Fletcher, V.K. Goyal, and P. Schniter. Hybrid approximate message passing with applications to structured sparsity. *Arxiv preprint arXiv:1111.2581*, Nov. 2011.

[84] S. Rangan, P. Schniter, and A. Fletcher. On the convergence of approximate message passing with arbitrary matrices. In *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pages 236–240, July 2014.

[85] G. Reeves and M. Gastpar. The sampling rate-distortion tradeoff for sparsity pattern recovery in compressed sensing. *IEEE Trans. Inform. Theory*, 58(5):3065–3092, May 2012.

[86] H. Rueda, D. Lau, and G. Arce. Multi-spectral compressive snapshot imaging using RGB image sensors. *to appear in Optics Express*, 2015.

[87] R. Schultz, T. Nielsen, J. Zavaleta, R. Ruch, R. Wyatt, and H. Garner. Hyperspectral imaging: A novel approach for microscopic analysis. *Cytometry*, 43(4):239–247, Apr. 2001.

[88] S. Sherman. A theorem on convex sets with applications. *Ann. Math. Stat.*, 26(4):763–767, Dec. 1955.

[89] S. Sherman. Non-mean-square error criteria. *IEEE Trans. Inf. Theory*, 4(3):125–126, Sept. 1958.

[90] E. Simoncelli and E. Adelson. Noise removal via Bayesian wavelet coring. In *Proc. Int. Conf. Image Process.*, volume 1, pages 379–382. IEEE, Sept. 1996.

[91] S. Som and P. Schniter. Compressive imaging using approximate message passing and a Markov-tree prior. *IEEE Trans. Signal Process.*, 60(7):3439–3448, July 2012.

[92] A. Soni and J. Haupt. Learning sparse representations for adaptive compressive sensing. In *IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, pages 2097–2100, Mar. 2012.

[93] J. Starck, F. Murtagh, and J. Fadili. *Sparse image and signal processing: Wavelets, curvelets, morphological diversity.* Cambridge University Press, 2010.

[94] C. Studer, W. Yin, and R. G. Baraniuk. Signal representations with minimum ℓ_∞-norm. In *Proc. 50th Allerton Conf. Commun., Control, Comput.*, Oct. 2012.

[95] M. Tabuchi, N. Yamane, and Y. Morikawa. Adaptive Wiener filter based on Gaussian mixture model for denoising chest X-ray CT image. In *Proc. IEEE SICE Annu. Conf.*, pages 682–689, Sept. 2007.

[96] J. Tan and D. Baron. Signal reconstruction in linear mixing systems with different error metrics. In *Inf. Theory Appl. Workshop*, Feb. 2013.

[97] J. Tan, D. Baron, and L. Dai. Signal estimation with low infinity-norm error by minimizing the mean p-norm error. In *Proc. IEEE 48th Conf. Inf. Sci. Syst.*, Mar. 2014.

[98] J. Tan, D. Baron, and L. Dai. Wiener filters in Gaussian mixture signal estimation with ℓ_∞-norm error. *IEEE Trans. Inf. Theory*, 60(10):6626–6635, Oct. 2014.

[99] J. Tan, D. Carmon, and D. Baron. Optimal estimation with arbitrary error metric in compressed sensing. In *Proc. IEEE Stat. Signal Process. Workshop (SSP)*, pages 588–591, Aug. 2012.

[100] J. Tan, D. Carmon, and D. Baron. Signal estimation with additive error metrics in compressed sensing. *IEEE Trans. Inf. Theory*, 60(1):150–158, Jan. 2014.

[101] J. Tan, Y. Ma, and D. Baron. Compressive imaging via approximate message passing with wavelet-based image denoising. In *Proc. IEEE Global Conf. Signal Inf. Process.*, Atlanta, GA, Dec. 2014.

[102] J. Tan, Y. Ma, and D. Baron. Compressive imaging via approximate message passing with image denoising. *IEEE Trans. Signal Process.*, 63(8):2085–2092, Apr. 2015.

[103] J. Tan, Y. Ma, H. Rueda, D. Baron, and G. Arce. Application of approximate message passing in coded aperture snapshot spectral imaging. In *Proc. IEEE Global Conf. Signal Inf. Process.*, to appear 2015.

[104] J. Tan, Y. Ma, H. Rueda, D. Baron, and G. Arce. Compressive hyperspectral imaging via approximate message passing. *arXiv preprint arXiv:1507.01248*, 2015. submitted.

[105] T. Tanaka. A statistical-mechanics approach to large-system analysis of CDMA multiuser detectors. *IEEE Trans. Inf. Theory*, 48(11):2888–2910, Nov. 2002.

[106] H. Taylor, S. Banks, and J. McCoy. Deconvolution with the ℓ_1 norm. *Geophysics*, 44(1):39–52, 1979.

[107] D. J. Thouless, P. W. Anderson, and R. G. Palmer. Solution of 'Solvable model of a spin glass'. *Philosophical Magazine*, 35:593–601, 1977.

[108] R. Tibshirani. Regression shrinkage and selection via the LASSO. *J. Royal Stat. Soc. Series B (Methodological)*, 58(1):267–288, 1996.

[109] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory*, 53(12):4655–4666, Dec. 2007.

[110] A. Tulino, G. Caire, S. Shamai, and S. Verdú. Support recovery with sparsely sampled free random matrices. In *IEEE Int. Symp. Inf. Theory*, pages 2328–2332, July 2011.

[111] J. Vila and P. Schniter. Expectation-maximization Bernoulli-Gaussian approximate message passing. In *Proc. IEEE 45th Asilomar Conf. Signals, Syst., and Comput.*, pages 799–803, Nov. 2011.

[112] J. Vila and P. Schniter. Expectation-maximization Gaussian-mixture approximate message passing. In *Proc. IEEE 46th Conf. Inf. Sci. Syst.*, Mar. 2012.

[113] J. Vila and P. Schniter. Expectation-maximization Gaussian-mixture approximate message passing. *IEEE Trans. Signal Process.*, 61(19):4658–4672, Oct. 2013.

[114] J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborova. Adaptive damping and mean removal for the generalized approximate message passing algorithm. *ArXiv preprint arXiv:1412.2005*, Dec. 2014.

[115] A. Wagadarikar, R. John, R. Willett, and D. Brady. Single disperser design for coded aperture snapshot spectral imaging. *Appl. Opt.*, 47(10):B44–B51, Apr. 2008.

[116] A. Wagadarikar, N. Pitsianis, X. Sun, and D. Brady. Spectral image estimation for coded aperture snapshot spectral imagers. In *Proc. SPIE*, page 707602, Sept. 2008.

[117] M.J. Wainwright. Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting. *IEEE Trans. Inf. Theory*, 55(12):5728–5741, Dec. 2009.

[118] W. Wang, M.J. Wainwright, and K. Ramchandran. Information-theoretic limits on sparse signal recovery: Dense versus sparse measurement matrices. *IEEE Trans. Inf. Theory*, 56(6):2967–2979, June 2010.

[119] A.R. Webb. *Statistical pattern recognition.* John Wiley & Sons Inc., 2002.

[120] N. Wiener. *Extrapolation, interpolation, and smoothing of stationary time series with engineering applications.* MIT press, 1949.

[121] R Willett, M Duarte, M Davenport, and R Baraniuk. Sparsity and structure in hyperspectral imaging: Sensing, reconstruction, and target detection. *IEEE Signal Process. Mag.*, 31(1):116–126, Jan. 2014.

[122] X. Yuan, T. Tsai, R. Zhu, P. Llull, D. Brady, and L. Carin. Compressive hyperspectral imaging with side information. *IEEE J. Sel. Topics Signal Process.*, PP(99):1, Mar. 2015.

[123] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin. Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Trans. Image Process.*, 21(1):130–144, Jan. 2012.

APPENDICES

Appendix A

Proof of Corollary 1

When

$$d_{\mathrm{AE}}(\widehat{x}_j, x_j) = |\widehat{x}_j - x_j|,$$

equations (4.13) or (4.14) solve for the MMAE estimand, $\widehat{\mathbf{x}}_{\mathrm{MMAE}}$. In order to find the minimum, we take the derivative of the expected function over $\widehat{x}_j$,

$$\left.\frac{d\, E(|\widehat{x}_j - x_j||\, q_j)}{d\widehat{x}_j}\right|_{\widehat{x}_j = \widehat{x}_{j,\mathrm{MMAE}}} = 0, \tag{A.1}$$

for each $j = \{1, 2, \ldots, N\}$. But

$$\begin{aligned} E(|\widehat{x}_j - x_j||q_j) &= \int_0^\infty \Pr(|\widehat{x}_j - x_j| > t|q_j)dt \\ = &\int_0^\infty \Pr(\widehat{x}_j - x_j > t|q_j)dt + \int_0^\infty \Pr(\widehat{x}_j - x_j < -t|q_j)dt \\ = &\int_{-\infty}^{\widehat{x}_j} \Pr(x_j < t_1|q_j)dt_1 + \int_{\widehat{x}_j}^\infty \Pr(x_j > t_2|q_j)dt_2, \end{aligned} \tag{A.2}$$

where changes of variables $t_1 = \widehat{x}_j - t$ and $t_2 = \widehat{x}_j + t$ are applied in (A.2). Using (A.1) and (A.2), we need

$$\Pr(x_j < \widehat{x}_{j,\mathrm{MMAE}}|q_j) - \Pr(x_j > \widehat{x}_{j,\mathrm{MMAE}}|q_j) = 0,$$

and thus $\widehat{x}_{j,\mathrm{MMAE}}$ is given as the median of the conditional statistics $f_{X_j|Q_j}(x_j|q_j)$,

$$\int_{-\infty}^{\widehat{x}_{j,\mathrm{MMAE}}} f_{X_j|Q_j}(x_j|q_j)dx_j = \int_{\widehat{x}_{j,\mathrm{MMAE}}}^{+\infty} f_{X_j|Q_j}(x_j|q_j)dx_j = \frac{1}{2}.$$

Then, the conditional mean absolute error is,

$$\begin{aligned}
& E[|\widehat{x}_{j,\text{MMAE}} - x_j||\, q_j] \\
= & \int_{-\infty}^{+\infty} |\widehat{x}_{j,\text{MMAE}} - x_j| f_{X_j|Q_j}(x_j|q_j) dx_j \\
= & \int_{-\infty}^{\widehat{x}_{j,\text{MMAE}}} (\widehat{x}_{j,\text{MMAE}} - x_j) f_{X_j|Q_j}(x_j|q_j) dx_j + \int_{\widehat{x}_{j,\text{MMAE}}}^{+\infty} (x_j - \widehat{x}_{j,\text{MMAE}}) f_{X_j|Q_j}(x_j|q_j) dx_j \\
= & \int_{-\infty}^{\widehat{x}_{j,\text{MMAE}}} (-x_j) f_{X_j|Q_j}(x_j|q_j) dx_j + \int_{\widehat{x}_{j,\text{MMAE}}}^{+\infty} x_j f_{X_j|Q_j}(x_j|q_j) dx_j.
\end{aligned}$$

Therefore, the MMAE for location j, $\text{MMAE}_j(f_{X_j}, \mu)$, is

$$\begin{aligned}
& \text{MMAE}_j(f_{X_j}, \mu) = E[|\widehat{x}_{j,\text{MMAE}} - x_j|] \\
= & \int_{-\infty}^{+\infty} E[|\widehat{x}_{j,\text{MMAE}} - x_j||\, q_j] f_{Q_j}(q_j) dq_j \\
= & \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{\widehat{x}_{j,\text{MMAE}}} (-x_j) f_{X_j|Q_j}(x_j|q_j) dx_j + \int_{\widehat{x}_{j,\text{MMAE}}}^{+\infty} x_j f_{X_j|Q_j}(x_j|q_j) dx_j \right) f_{Q_j}(q_j) dq_j.
\end{aligned}$$

We note in passing that the integrations can be evaluated numerically in an implementation.

Because the input $\mathbf{x}$ is i.i.d., and the decoupled scalar channels have the same parameter μ, the values of MMAE_j for all $j \in \{1, 2, \ldots, N\}$ are the same, and the overall MMAE is

$$\text{MMAE}(f_{\mathbf{X}}, \mu) = N \cdot \text{MMAE}_j(f_{X_j}, \mu).$$

Appendix B

Proof of Corollary 2

Similar to the idea of giving a limit on support recovery error rate [110], we derive the MMSuE limit for the case where the input is real-valued and the matrix $\mathbf{A}$ is rectangular ($M < N$). In the scalar Gaussian channel (3.2), we factor the sparse Gaussian input X_j into $X_j = U_j \cdot B_j$, where $U_j \sim \mathcal{N}(0, \sigma^2)$ and $B_j \sim$ Bernoulli(p), i.e., $\Pr(B_j = 1) = p = 1 - \Pr(B_j = 0)$. The support recovery problem is the task of finding the maximum a-posteriori (MAP) estimation of B_j.

For our estimation algorithm, the conditional expectation of support recovery error is,

$$E\left[d(\widehat{x}_j, x_j)|q_j\right] = \begin{cases} \Pr(B_j = 1|q_j) & \text{if } \widehat{b}_j = 0 \text{ and } b_j = 1 \\ \Pr(B_j = 0|q_j) & \text{if } \widehat{b}_j = 1 \text{ and } b_j = 0 \\ 0 & \text{if } \widehat{b}_j = 0 \text{ and } b_j = 0 \\ 0 & \text{if } \widehat{b}_j = 1 \text{ and } b_j = 1 \end{cases}.$$

The estimand $\widehat{b}_{j,\text{opt}}$ minimizes $E\left[d(\widehat{x}_j, x_j)|q_j\right]$, which implies

$$\widehat{b}_{j,\text{opt}} = \begin{cases} 0 & \text{if } \Pr(B_j = 1|q_j) \leq \Pr(B_j = 0|q_j) \\ 1 & \text{if } \Pr(B_j = 1|q_j) > \Pr(B_j = 0|q_j) \end{cases}. \tag{B.1}$$

It is easy to see that $f_{Q_j|B_j}(q_j|0) \sim \mathcal{N}(0,\mu)$ and $f_{Q_j|B_j}(q_j|1) \sim \mathcal{N}(0,\sigma^2+\mu)$. Then,

$$\begin{aligned} \Pr(B_j=1|q_j) &= \frac{f_{Q_j|B_j}(q_j|1)\Pr(B_j=1)}{f_{Q_j}(q_j)} \\ &= \frac{f_{Q_j|B_j}(q_j|1)\Pr(B_j=1)}{\sum_{b_j=0,1} f_{Q_j|B_j}(q_j|b_j)\Pr(B_j=b_j)} \\ &= \frac{1}{1+\frac{1-p}{p}\sqrt{\sigma^2/\mu+1}\exp\Big(-\frac{q_j^2}{2}\cdot\frac{\sigma^2/\mu}{\sigma^2+\mu}\Big)}, \end{aligned} \tag{B.2}$$

and similarly

$$\begin{aligned} \Pr(B_j=0|q_j) &= \frac{f_{Q_j|B_j}(q_j|0)\Pr(B_j=0)}{f_{Q_j}(q_j)} \\ &= \frac{\frac{1-p}{p}\sqrt{\sigma^2/\mu+1}\exp\Big(-\frac{q_j^2}{2}\cdot\frac{\sigma^2/\mu}{\sigma^2+\mu}\Big)}{1+\frac{1-p}{p}\sqrt{\sigma^2/\mu+1}\exp\Big(-\frac{q_j^2}{2}\cdot\frac{\sigma^2/\mu}{\sigma^2+\mu}\Big)}. \end{aligned} \tag{B.3}$$

Therefore, $\Pr(B_j=1|q_j) > \Pr(B_j=0|q_j)$ implies

$$q_j^2 > \tau = 2\cdot\frac{\sigma^2+\mu}{\sigma^2/\mu}\ln\left(\frac{(1-p)\sqrt{\sigma^2/\mu+1}}{p}\right),$$

and vice versa. We can rewrite (B.1) as,

$$\widehat{b}_{j,\text{opt}} = \begin{cases} 0 & \text{if } q_j^2 \le \tau \\ 1 & \text{if } q_j^2 > \tau \end{cases}.$$

By averaging over the range of Q_j, we get the overall MMSuE,

$$\begin{aligned} & \text{MMSuE}(f_{\mathbf{X}},\mu) = N\cdot E[d_{j,\text{support}}(\widehat{x}_j,x_j)] \\ = \; & N\int E\left(d_{j,\text{support}}(\widehat{x}_j,x_j)|q_j\right) f_{Q_j}(q_j)dq_j \\ = \; & N\int_{q_j^2>\tau}\Pr(B_j=0|q_j)f_{Q_j}(q_j)dq_j + N\int_{q_j^2\le\tau}\Pr(B_j=1|q_j)f_{Q_j}(q_j)dq_j \\ = \; & N\cdot\Pr(B_j=0,q_j^2>\tau) + N\cdot\Pr(B_j=1,q_j^2\le\tau) \\ = \; & N\cdot\Pr(q_j^2>\tau|B_j=0)\Pr(B_j=0) + N\cdot\Pr(q_j^2\le\tau|B_j=1)\Pr(B_j=1) \\ = \; & N(1-p)\cdot\text{erfc}\left(\sqrt{\frac{\tau}{2\mu}}\right) + Np\cdot\text{erf}\left(\sqrt{\frac{\tau}{2(\sigma^2+\mu)}}\right). \end{aligned}$$

Appendix C

Proof of Corollary 3

We use the same variables U_j and B_j as defined in Appendix B. For $d_{\text{w\_support}}$ (3.9), its conditional expectation is

$$E\left[d_{\text{w\_support}}(\widehat{x}_j, x_j)|q_j\right] = \begin{cases} (1-\beta)\cdot\Pr(B_j=1|q_j) & \text{if } \widehat{b}_j = 0 \text{ and } b_j = 1 \\ \beta\cdot\Pr(B_j=0|q_j) & \text{if } \widehat{b}_j = 1 \text{ and } b_j = 0 \\ 0 & \text{if } \widehat{b}_j = 0 \text{ and } b_j = 0 \\ 0 & \text{if } \widehat{b}_j = 1 \text{ and } b_j = 1 \end{cases}.$$

The estimand $\widehat{b}_{j,\text{opt}}$ minimizes $E\left[d_{\text{w\_support}}(\widehat{x}_j, x_j)|q_j\right]$, which implies

$$\widehat{b}_{j,\text{opt}} = \begin{cases} 0 & \text{if } (1-\beta)\cdot\Pr(B_j=1|q_j) \leq \beta\cdot\Pr(B_j=0|q_j) \\ 1 & \text{if } (1-\beta)\cdot\Pr(B_j=1|q_j) > \beta\cdot\Pr(B_j=0|q_j) \end{cases}. \tag{C.1}$$

Plugging (B.2) and (B.3) into (C.1), we get that

$$\widehat{b}_{j,\text{opt}} = \begin{cases} 0 & \text{if } q_j^2 \leq \tau' \\ 1 & \text{if } q_j^2 > \tau' \end{cases},$$

where

$$q_j^2 > \tau' = 2\cdot\frac{\sigma^2+\mu}{\sigma^2/\mu}\ln\left(\frac{\beta(1-p)\sqrt{\sigma^2/\mu+1}}{(1-\beta)p}\right).$$

Therefore, the minimum mean weighted-support error function $\text{MMWSE}(f_{\mathbf{X}}, \mu)$ is,

$$\begin{aligned}
&\text{MMWSE}(f_{\mathbf{X}}, \mu) = N \cdot E[d_{j,\text{w\_support}}(\widehat{x}_j, x_j)] \\
= \;& N \int E\left(d_{j,\text{w\_support}}(\widehat{x}_j, x_j)|q_j\right) f_{Q_j}(q_j)dq_j \\
= \;& N \int_{q_j^2 > \tau'} \beta \Pr(B_j = 0|q_j) f_{Q_j}(q_j)dq_j + N \int_{q_j^2 \leq \tau'} (1-\beta)\Pr(B_j = 1|q_j) f_{Q_j}(q_j)dq_j \\
= \;& N\beta(1-p) \cdot \text{erfc}\left(\sqrt{\frac{\tau'}{2\mu}}\right) + N(1-\beta)p \cdot \text{erf}\left(\sqrt{\frac{\tau'}{2(\sigma^2+\mu)}}\right).
\end{aligned}$$

The false positive rate is

$$\begin{aligned}
\Pr(\widehat{x}_j \neq 0|x_j = 0) &= \Pr(q_j^2 > \tau'|x_j = 0) \\
&= \Pr(q_j^2 > \tau'|B_j = 0) \\
&= \text{erfc}\left(\sqrt{\frac{\tau'}{2\mu}}\right),
\end{aligned}$$

and the false negative rate is

$$\begin{aligned}
\Pr(\widehat{x}_j = 0|x_j \neq 0) &= \Pr(q_j^2 \leq \tau'|x_j \neq 0) \\
&= \Pr(q_j^2 \leq \tau'|B_j = 1) \\
&= \text{erf}\left(\sqrt{\frac{\tau'}{2(\sigma^2+\mu)}}\right).
\end{aligned}$$

Appendix D

Proof of Theorem 1

Two error patterns: We begin by defining two error patterns. Consider parallel Gaussian channels (4.3), where the input signal $x_i \sim s \cdot \mathcal{N}(\mu_x, \sigma_x^2) + (1-s) \cdot \delta(x_i)$ for some s, and the noise $u_i \sim \mathcal{N}(0, \sigma_u^2)$. The Wiener filter (linear estimator) for the Bernoulli-Gaussian input is $\widehat{\mathbf{x}}_{\text{W,BG}} = \frac{\sigma_x^2}{\sigma_x^2+\sigma_u^2} \cdot (\mathbf{w} - \mu_x) + \mu_x$. Let $\mathcal{I}$ denote the index set where $x_i \sim \mathcal{N}(\mu_x, \sigma_x^2)$, and let $\mathcal{J}$ denote the index set where $x_j \sim \delta(x_j)$. We define two types of error patterns: (i) for

$$i \in \mathcal{I} \triangleq \left\{ i : x_i \sim \mathcal{N}\left(\mu_x, \sigma_x^2\right) \right\},$$

the error is

$$e_i \triangleq \widehat{x}_{\text{W,BG},i} - x_i = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_u^2} \cdot (w_i - \mu_x) + \mu_x - x_i \sim \mathcal{N}\left(0, \frac{\sigma_x^2 \sigma_u^2}{\sigma_x^2 + \sigma_u^2}\right),$$

where we remind readers that $\widehat{x}_{\text{W,BG},i}$ denotes the i-th component of the vector $\widehat{\mathbf{x}}_{\text{W,BG}}$ in (4.7); and (ii) for

$$j \in \mathcal{J} \triangleq \{ j : x_j \sim \delta(x_j) \},$$

the error is

$$\widetilde{e}_j \triangleq \widehat{x}_{\text{W,BG},j} - x_j = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_u^2} \cdot (w_i - \mu_x) + \mu_x - 0 \sim \mathcal{N}\left(\frac{\sigma_u^2}{\sigma_x^2 + \sigma_u^2}\mu_x, \frac{\sigma_x^4 \sigma_u^2}{(\sigma_x^2 + \sigma_u^2)^2}\right).$$

Maximum of error patterns: Let us compare $\max_{i \in \mathcal{I}} |e_i|$ and $\max_{j \in \mathcal{J}} |\widetilde{e}_j|$.

Lemma 1 *Suppose g_i is an i.i.d. Gaussian sequence of length N, $g_i \sim \mathcal{N}(\mu, \sigma^2)$ for $i \in \{1, 2, \ldots, N\}$, then $\frac{\max_{1 \le i \le N} |g_i|}{\sqrt{2\sigma^2 \cdot \ln(N)}}$ converges to 1 in probability. That is,*

$$\lim_{N \to \infty} \Pr\left(\left| \frac{\max_{1 \le i \le N} |g_i|}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1 \right| < \Delta \right) = 1, \tag{D.1}$$

for any $\Delta > 0$.

Lemma 1 is proved in Section F.

Before applying Lemma 1, we define a set A_ϵ of possible inputs $\mathbf{x}$ such that the numbers of components in the sets $\mathcal{I}$ and $\mathcal{J}$ both go to infinity as $N \to \infty$,

$$A_\epsilon \triangleq \left\{ \mathbf{x} : \left| \frac{|\mathcal{I}|}{N} - s \right| < \epsilon \right\}, \tag{D.2}$$

where $\epsilon > 0$ and $\epsilon \to 0$ (namely, $\epsilon \to 0^+$) as a function of signal dimension N, and $|\mathcal{I}|$ denotes the cardinality of the set $\mathcal{I}$. The definition of A_ϵ suggests that $\left| \frac{|\mathcal{J}|}{N} - (1-s) \right| < \epsilon$ and $|\mathcal{I}|+|\mathcal{J}| = N$. Therefore, if $\mathbf{x} \in A_\epsilon$, then $|\mathcal{I}|, |\mathcal{J}| \to \infty$ as $N \to \infty$.

Now we are ready to evaluate $\max_{i\in\mathcal{I}} |e_i|$ and $\max_{j\in\mathcal{J}} |\widetilde{e}_j|$. For i.i.d. Gaussian random variables $e_i \sim \mathcal{N}(0, \frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2})$, where $i \in \mathcal{I}$, the equality (D.1) in Lemma 1 becomes

$$\lim_{N\to\infty} \Pr\left(\left| \frac{\max_{i\in\mathcal{I}} |e_i|}{\sqrt{2 \cdot \frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2} \cdot \ln(|\mathcal{I}|)}} - 1 \right| < \Delta \middle| \mathbf{x} \in A_\epsilon \right) = 1, \tag{D.3}$$

for any $\Delta > 0$. For i.i.d. Gaussian random variables $\widetilde{e}_j$, where $j \in \mathcal{J}$, the equality (D.1) becomes

$$\lim_{N\to\infty} \Pr\left(\left| \frac{\max_{j\in\mathcal{J}} |\widetilde{e}_j|}{\sqrt{2 \cdot \frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2} \cdot \ln(|\mathcal{J}|)}} - 1 \right| < \Delta \middle| \mathbf{x} \in A_\epsilon \right) = 1, \tag{D.4}$$

for any $\Delta > 0$.

Equations (D.3) and (D.4) suggest that

$$\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}} |e_i|}{\sqrt{2 \cdot \frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2} \cdot \ln(|\mathcal{I}|)}} \middle| \mathbf{x} \in A_\epsilon \right] = 1$$

and

$$\lim_{N\to\infty} E\left[\frac{\max_{j\in\mathcal{J}} |\widetilde{e}_j|}{\sqrt{2 \cdot \frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2} \cdot \ln(|\mathcal{J}|)}} \middle| \mathbf{x} \in A_\epsilon \right] = 1,$$

which yield

$$\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}} |e_i|}{\sqrt{\ln(N)}} \middle| \mathbf{x} \in A_\epsilon \right] = \lim_{N\to\infty} \sqrt{2 \cdot \frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2} \cdot \frac{\ln(|\mathcal{I}|)}{\ln(N)}} \tag{D.5}$$

and

$$\lim_{N\to\infty} E\left[\frac{\max_{j\in\mathcal{J}} |\widetilde{e}_j|}{\sqrt{\ln(N)}} \middle| \mathbf{x} \in A_\epsilon \right] = \lim_{N\to\infty} \sqrt{2 \cdot \frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2} \cdot \frac{\ln(|\mathcal{J}|)}{\ln(N)}}. \tag{D.6}$$

According to the definition of A_ϵ in (D.2), where s is a constant, and $\epsilon \to 0^+$,

$$N(s-\epsilon) < |\mathcal{I}| < N(s+\epsilon) \quad \text{and} \quad N(1-s-\epsilon) < |\mathcal{J}| < N(1-s+\epsilon), \tag{D.7}$$

and thus

$$\lim_{N\to\infty} \sqrt{\frac{\ln(|\mathcal{I}|)}{\ln(N)}} = 1 \quad \text{and} \quad \lim_{N\to\infty} \sqrt{\frac{\ln(|\mathcal{J}|)}{\ln(N)}} = 1. \tag{D.8}$$

Finally, equations (D.5) and (D.6) become

$$\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}}|e_i|}{\sqrt{\ln(N)}}\middle|\mathbf{x}\in A_\epsilon\right] = \sqrt{2\cdot\frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2}} \tag{D.9}$$

and

$$\lim_{N\to\infty} E\left[\frac{\max_{j\in\mathcal{J}}|\widetilde{e}_j|}{\sqrt{\ln(N)}}\middle|\mathbf{x}\in A_\epsilon\right] = \sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}}. \tag{D.10}$$

Combining (D.3) and (D.4),

$$\lim_{N\to\infty} \Pr\left(\frac{1-\Delta}{1+\Delta} < \frac{\max_{i\in\mathcal{I}}|e_i|}{\max_{j\in\mathcal{J}}|\widetilde{e}_j|}\cdot\frac{\sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}\cdot\ln(|\mathcal{J}|)}}{\sqrt{2\cdot\frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2}\cdot\ln(|\mathcal{I}|)}} < \frac{1+\Delta}{1-\Delta}\middle|\mathbf{x}\in A_\epsilon\right) = 1. \tag{D.11}$$

Note that

$$\sqrt{\frac{\ln(N)+\ln(1-s-\epsilon)}{\ln(N)+\ln(s+\epsilon)}} = \sqrt{\frac{\ln(N(1-s-\epsilon))}{\ln(N(s+\epsilon))}} < \sqrt{\frac{\ln(|\mathcal{J}|)}{\ln(|\mathcal{I}|)}} < \sqrt{\frac{\ln(N(1-s+\epsilon))}{\ln(N(s-\epsilon))}} = \sqrt{\frac{\ln(N)+\ln(1-s+\epsilon)}{\ln(N)+\ln(s-\epsilon)}}.$$

Then the following limit holds,

$$\lim_{N\to\infty}\sqrt{\frac{\ln(|\mathcal{J}|)}{\ln(|\mathcal{I}|)}} = 1.$$

We can write the above limit in probabilistic form,

$$\lim_{N\to\infty} \Pr\left(\left|\sqrt{\frac{\ln(|\mathcal{J}|)}{\ln(|\mathcal{I}|)}} - 1\right| < \Delta\middle|\mathbf{x}\in A_\epsilon\right) = 1, \tag{D.12}$$

for any $\Delta > 0$. Because of the logarithms in (D.12), the ratio $\frac{\sqrt{2\cdot\ln(|\mathcal{J}|)}}{\sqrt{2\cdot\ln(|\mathcal{I}|)}}$ is sufficiently close to 1 as N is astronomically large. This is why we point out in Section 4.7 that the asymptotic results in this correspondence might be impractical. Plugging (D.12) into (D.11),

$$\lim_{N\to\infty} \Pr\left(\frac{1-\Delta}{(1+\Delta)^2} \cdot \sqrt{\frac{\sigma_x^2+\sigma_u^2}{\sigma_x^2}} < \frac{\max_{i\in\mathcal{I}}|e_i|}{\max_{j\in\mathcal{J}}|\widetilde{e}_j|} < \frac{1+\Delta}{(1-\Delta)^2} \cdot \sqrt{\frac{\sigma_x^2+\sigma_u^2}{\sigma_x^2}} \right| \mathbf{x}\in A_\epsilon \Bigg) = 1. \quad \text{(D.13)}$$

Equation (D.13) holds for any $\Delta > 0$. We note that $\sqrt{\frac{\sigma_x^2+\sigma_u^2}{\sigma_x^2}} > 1$, and thus $\frac{1-\Delta}{(1+\Delta)^2} \cdot \sqrt{\frac{\sigma_x^2+\sigma_u^2}{\sigma_x^2}} > 1$ for sufficiently small Δ. Therefore,

$$\begin{aligned} & \lim_{N\to\infty} \Pr\left(\frac{\max_{i\in\mathcal{I}}|e_i|}{\max_{j\in\mathcal{J}}|\widetilde{e}_j|} > 1 \right| \mathbf{x}\in A_\epsilon \Bigg) \\ = & \lim_{N\to\infty} \Pr\left(\frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|} > 1 \right| \mathbf{x}\in A_\epsilon \Bigg) \\ = & \ 1, \end{aligned} \quad \text{(D.14)}$$

and

$$\lim_{N\to\infty} \Pr\left(\frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|} \le 1 \right| \mathbf{x}\in A_\epsilon \Bigg) = 0.$$

Mean ℓ_∞-norm error: The road map for the remainder of the proof is to first show that when $\mathbf{x}\in A_\epsilon$ the Wiener filter is asymptotically optimal for expected ℓ_∞-norm error, and then show that $\Pr(\mathbf{x}\in A_\epsilon)$ is arbitrarily close to 1.

In order to utilize equations (D.9) and (D.10), we normalize the quantities in the following derivations by $\sqrt{\ln(N)}$ so that every term is bounded.

$$\begin{aligned} & \lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty | \, \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}} \\ = & \lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\sqrt{\ln(N)}} \right| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|} > 1 \Bigg] \\ & \times \Pr\left(\frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|} > 1 \right| \mathbf{x}\in A_\epsilon \Bigg) \\ + & \lim_{N\to\infty} E\left[\frac{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|}{\sqrt{\ln(N)}} \right| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|} \le 1 \Bigg] \\ & \times \Pr\left(\frac{\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}}|x_j-\widehat{x}_{\text{W,BG},j}|} \le 1 \right| \mathbf{x}\in A_\epsilon \Bigg). \end{aligned} \quad \text{(D.15)}$$

Let us now verify that the second term in (D.15) equals 0. In fact, the following derivations hold from (D.4) and (D.14),

$$\begin{aligned} 1 &= \lim_{N\to\infty} \Pr\left(\left| \frac{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,GB},j}|}{\sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}\cdot\ln(|\mathcal{J}|)}} - 1 \right| < \Delta \middle| \mathbf{x}\in A_\epsilon \right) \\ &= \lim_{N\to\infty} \Pr\left(\left| \frac{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,GB},j}|}{\sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}\cdot\ln(|\mathcal{J}|)}} - 1 \right| < \Delta \middle| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,BG},j}|} > 1 \right). \end{aligned}$$

Therefore,

$$\lim_{N\to\infty} E\left[\frac{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,GB},j}|}{\sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}\cdot\ln(|\mathcal{J}|)}} \middle| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,BG},j}|} > 1 \right] = 1,$$

which yields (following similar derivations of (D.6) and (D.10))

$$\lim_{N\to\infty} E\left[\frac{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,GB},j}|}{\sqrt{\ln(N)}} \middle| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,BG},j}|} > 1 \right] = \sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}}.$$

Therefore, the second term in (D.15) equals $\sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}}\times 0 = 0$, and equation (D.15) becomes

$$\begin{aligned} &\lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty | \, \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}} \\ =\ &\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\sqrt{\ln(N)}} \middle| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,BG},j}|} > 1 \right] \\ =\ &\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\sqrt{\ln(N)}} \middle| \mathbf{x}\in A_\epsilon \right] \\ -\ &\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\sqrt{\ln(N)}} \middle| \mathbf{x}\in A_\epsilon, \frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,BG},j}|} \le 1 \right] \\ &\times \Pr\left(\frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\max_{j\in\mathcal{J}} |x_j - \widehat{x}_{\text{W,BG},j}|} \le 1 \middle| \mathbf{x}\in A_\epsilon \right) \\ =\ &\lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}} |x_i - \widehat{x}_{\text{W,BG},i}|}{\sqrt{\ln(N)}} \middle| \mathbf{x}\in A_\epsilon \right]. \qquad \text{(D.16)} \end{aligned}$$

Equation (D.16) shows that the maximum absolute error of the Wiener filter relates to the Gaussian-distributed components of $\mathbf{x}$.

Optimality of the Wiener filter: It has been shown by Sherman [88, 89] that, for parallel Gaussian channels with an i.i.d. Gaussian input $\mathbf{x}$, if an error metric function $d(\mathbf{x},\widehat{\mathbf{x}})$ relating $\mathbf{x}$ and its estimate $\widehat{\mathbf{x}}$ is convex, then the Wiener filter is optimal for that error metric. The ℓ_∞-norm is convex, and therefore, for any estimator $\widehat{\mathbf{x}}$,

$$\begin{aligned}
& E\left[\|\mathbf{x}-\widehat{\mathbf{x}}\|_\infty \middle| \mathbf{x}\in A_\epsilon\right] \\
= \; & E\left[\max_{i\in\mathcal{I}\cup\mathcal{J}} |x_i-\widehat{x}_i| \middle| \mathbf{x}\in A_\epsilon\right] \\
\geq \; & E\left[\max_{i\in\mathcal{I}} |x_i-\widehat{x}_i| \middle| \mathbf{x}\in A_\epsilon\right] \\
\geq \; & E\left[\max_{i\in\mathcal{I}} |x_i-\widehat{x}_{\mathrm{W,BG},i}| \middle| \mathbf{x}\in A_\epsilon\right]. \qquad \text{(D.17)}
\end{aligned}$$

The inequality (D.17) holds, because the set $\{x_i : i\in\mathcal{I}\}$ only contains the i.i.d. Gaussian components of $\mathbf{x}$, and the Wiener filter is optimal for ℓ_∞-norm error when the input signal is i.i.d. Gaussian. The inequality (D.17) holds for any signal length N, and thus it holds when $N\to\infty$ and we divide both sides by $\sqrt{\ln(N)}$,

$$\begin{aligned}
0 \;\leq\; & \lim_{N\to\infty}\left(\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}\|_\infty | \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}} - \frac{E\left[\max_{i\in\mathcal{I}}|x_i-\widehat{x}_{\mathrm{W,BG},i}| | \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\right) \\
=\; & \lim_{N\to\infty}\left(\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}\|_\infty | \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}} - \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty | \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\right), \qquad \text{(D.18)}
\end{aligned}$$

where the last step in (D.18) is justified by the derivation of (D.16). Equation (D.18) also holds for $\widehat{\mathbf{x}}=\widehat{\mathbf{x}}_{\ell_\infty}$,

$$\lim_{N\to\infty}\left(\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty | \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}} - \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty | \mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\right) \geq 0. \qquad \text{(D.19)}$$

Typical set: Let us now evaluate $\Pr(\mathbf{x}\in A_\epsilon)$. The set A_ϵ only considers whether the components in $\mathbf{x}$ are Gaussian or zero, and so we introduce a binary vector $\widetilde{\mathbf{x}}\in\mathbb{R}^N$, where $\widetilde{x}_i = \mathbf{1}_{\{x_i\sim\mathcal{N}(\mu_x\sigma_x^2)\}}$ and $\mathbf{1}_{\{\cdot\}}$ is the indicator function. That is, $\widetilde{x}_i=1$ if x_i is Gaussian, and else $\widetilde{x}_i=0$. The sequence $\widetilde{\mathbf{x}}\triangleq\{\widetilde{x}_1,\widetilde{x}_2,\ldots,\widetilde{x}_N\}$ is called a *typical sequence* ([28], page 59), if it satisfies

$$2^{-N(H(\widetilde{\mathbf{X}})+\delta)} \leq \Pr(\widetilde{x}_1,\widetilde{x}_2,\ldots,\widetilde{x}_N) \leq 2^{-N(H(\widetilde{\mathbf{X}})-\delta)}, \qquad \text{(D.20)}$$

for some $\delta > 0$, where $H(\widetilde{\mathbf{X}})$ denotes the binary entropy [28] of the sequence $\{\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_N\}$. The set A_ϵ is then called a *typical set* [28], and

$$\Pr(\mathbf{x} \in A_\epsilon) > 1 - \delta. \tag{D.21}$$

We highlight that the inequalities (D.20) and (D.21) both hold when $\delta \to 0^+$ as a function of N.

In our problem setting where $\Pr(\widetilde{x}_i = 1) = \Pr(x_i \sim \mathcal{N}(\mu_x, \sigma_x^2)) = s$, the entropy of the sequence $\{\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_N\}$ is

$$H(\widetilde{\mathbf{X}}) = -s \log_2(s) - (1-s)\log_2(1-s), \tag{D.22}$$

and the probability of the sequence $\{\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_N\}$ is

$$\Pr(\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_N) = s^{|\mathcal{I}|} \cdot (1-s)^{|\mathcal{J}|}. \tag{D.23}$$

Plugging (D.7), (D.22), and (D.23) into (D.20), the value of δ can be computed,

$$\delta = \epsilon \left| \log_2 \left(\frac{s}{1-s} \right) \right|, \tag{D.24}$$

for $0 < s < 1$ and $s \neq 0.5$. That is,

$$\Pr(\mathbf{x} \in A_\epsilon) > 1 - \delta = 1 - \epsilon \left| \log_2 \left(\frac{s}{1-s} \right) \right|. \tag{D.25}$$

Finally, we compare $E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{WB,G}}\|_\infty\right]$ with $E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]$, where $\widehat{\mathbf{x}}_{\ell_\infty}$ satisfies (4.5), i.e., the estimate $\widehat{\mathbf{x}}_{\ell_\infty}$ is optimal for minimizing the mean ℓ_∞-norm error of reconstruction. By definition,

$$\lim_{N\to\infty} \left(\frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}} - \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty\right]}{\sqrt{\ln(N)}} \right) \leq 0,$$

but we already proved in (D.19) that

$$\lim_{N\to\infty} \left(\frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty | \, \mathbf{x} \in A_\epsilon\right]}{\sqrt{\ln(N)}} - \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty | \, \mathbf{x} \in A_\epsilon\right]}{\sqrt{\ln(N)}} \right) \geq 0,$$

and thus

$$\lim_{N\to\infty} \left(\frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty | \, \mathbf{x} \notin A_\epsilon\right]}{\sqrt{\ln(N)}} - \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty | \, \mathbf{x} \notin A_\epsilon\right]}{\sqrt{\ln(N)}} \right) \leq 0. \tag{D.26}$$

We know that $\Pr(\mathbf{x} \notin A_\epsilon) < \delta$ from (D.25). To complete the proof, it suffices to show that, when $\mathbf{x} \notin A_\epsilon$, the subtraction (D.26) is bounded. When $\mathbf{x} \notin A_\epsilon$, there are 3 cases for the possible values of $|\mathcal{I}|$ and $|\mathcal{J}|$:

- Case 1: $|\mathcal{I}|, |\mathcal{J}| \to \infty$, but (D.8) may not hold.
- Case 2: $|\mathcal{J}| \to \infty$ but $|\mathcal{I}| \not\to \infty$.
- Case 3: $|\mathcal{I}| \to \infty$ but $|\mathcal{J}| \not\to \infty$.

We observe that equations (D.9) and (D.10) are derived from (D.5), (D.6), and (D.8). In Case 1, similar equalities to (D.5) and (D.6) hold,

$$\lim_{N\to\infty} E\left[\left.\frac{\max_{i\in\mathcal{I}}|e_i|}{\sqrt{\ln(N)}}\right| \text{Case 1 of } \mathbf{x} \notin A_\epsilon\right] = \lim_{N\to\infty}\sqrt{2\cdot\frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2}\cdot\frac{\ln(|\mathcal{I}|)}{\ln(N)}} \leq \sqrt{2\cdot\frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2}}$$

and

$$\lim_{N\to\infty} E\left[\left.\frac{\max_{j\in\mathcal{J}}|\widetilde{e}_j|}{\sqrt{\ln(N)}}\right| \text{Case 1 of } \mathbf{x} \notin A_\epsilon\right] = \lim_{N\to\infty}\sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}\cdot\frac{\ln(|\mathcal{J}|)}{\ln(N)}} \leq \sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}}.$$

Therefore, the value of $\lim_{N\to\infty} E\left[\left.\frac{\|\mathbf{x}-\widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty}{\sqrt{\ln(N)}}\right| \text{Case 1 of } \mathbf{x} \notin A_\epsilon\right]$ is bounded.

In Case 2, it is obvious that $\lim_{N\to\infty} E\left[\left.\frac{\max_{i\in\mathcal{I}}|e_i|}{\sqrt{\ln(N)}}\right| \text{Case 2 of } \mathbf{x} \notin A_\epsilon\right]$ is bounded because $|\mathcal{I}| \not\to \infty$, while $\lim_{N\to\infty} E\left[\left.\frac{\max_{j\in\mathcal{J}}|\widetilde{e}_j|}{\sqrt{\ln(N)}}\right| \text{Case 2 of } \mathbf{x} \notin A_\epsilon\right]$ is bounded because $|\mathcal{J}| \leq N$, and

$$\lim_{N\to\infty} E\left[\left.\frac{\max_{j\in\mathcal{J}}|\widetilde{e}_j|}{\sqrt{\ln(N)}}\right| \text{Case 2 of } \mathbf{x} \notin A_\epsilon\right] \leq \sqrt{2\cdot\frac{\sigma_x^4\sigma_u^2}{(\sigma_x^2+\sigma_u^2)^2}}.$$

The analysis for Case 3 is similar to that of Case 2.

Therefore, we have shown that $\lim_{N\to\infty} E\left[\left.\frac{\|\mathbf{x}-\widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty}{\sqrt{\ln(N)}}\right| \mathbf{x} \notin A_\epsilon\right]$ is bounded.

By (D.26), $\lim_{N\to\infty} \frac{E[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon]}{\sqrt{\ln(N)}}$ is bounded above by $\lim_{N\to\infty} E\left[\left.\frac{\|\mathbf{x}-\widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty}{\sqrt{\ln(N)}}\right| \mathbf{x} \notin A_\epsilon\right]$. Hence,

$$\lim_{N\to\infty}\left(\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\text{W,BG}}\|_\infty|\,\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}} - \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\,\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\right) = c$$

is bounded, where $c > 0$ is a constant.

Therefore,

$$
\begin{aligned}
& \lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty\right]}{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} \\
= \quad & \lim_{N\to\infty} \frac{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty\right]}{\sqrt{\ln(N)}}}{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}}} \\
= \quad & \lim_{N\to\infty} \frac{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty|\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right)+\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)}{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right)+\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)} \\
= \quad & \lim_{N\to\infty} \frac{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty|\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right)+\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)}{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right)+\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)} \\
& + \lim_{N\to\infty} \frac{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)-\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)}{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right)+\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)} \\
\leq \quad & 1+\lim_{N\to\infty} \frac{\left(\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}-\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\right)\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right)}{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}}} \\
< \quad & 1+\lim_{N\to\infty} \frac{c\cdot\delta}{\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}}}. \qquad \text{(D.27)}
\end{aligned}
$$

In (D.27), the value of $\lim_{N\to\infty}\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}}$ is bounded below because of (D.19),

$$
\begin{aligned}
& \lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}} \\
= \quad & \lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\,\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right)+\lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty|\,\mathbf{x}\notin A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\notin A_\epsilon\right) \\
\geq \quad & \lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty|\,\mathbf{x}\in A_\epsilon\right]}{\sqrt{\ln(N)}}\cdot \Pr\left(\mathbf{x}\in A_\epsilon\right) \\
> \quad & \sqrt{2\cdot\frac{\sigma_x^2\sigma_u^2}{\sigma_x^2+\sigma_u^2}}\cdot(1-\delta).
\end{aligned}
$$

On the other hand, whether the value of $\frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]}{\sqrt{\ln(N)}}$ is bounded above or not, the second term in (D.27) is always arbitrarily small because δ is arbitrarily small, and thus (D.27) is equivalent to

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty\right]}{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} < 1+\delta,$$

where $\delta \to 0^+$ as a function of N. Finally, because $\widehat{\mathbf{x}}_\infty$ is the optimal estimator for ℓ_∞-norm error,

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty\right]}{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} \geq 1.$$

Therefore,

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\mathrm{W,BG}}\|_\infty\right]}{E\left[\|\mathbf{x}-\widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} = 1,$$

which completes the proof.

Appendix E

Proof of Theorem 2

The road map of the proof of Theorem 2 is the same as that of Theorem 1.

K **error patterns:** The input signal of the parallel Gaussian channels (4.3) is generated by an i.i.d. Gaussian mixture source (4.1), and suppose without loss of generality that $\sigma_1^2 = \max_{k\in\{1,2,\dots,K\}} \sigma_k^2$. The Wiener filter is $\widehat{\mathbf{x}}_{\text{W,GM}} = \frac{\sigma_1^2}{\sigma_1^2+\sigma_u^2} \cdot (\mathbf{w} - \mu_1) + \mu_1 = \frac{\sigma_1^2\mathbf{w}+\sigma_u^2\mu_1}{\sigma_1^2+\sigma_u^2}$. Let $\mathcal{I}_k$ denote the index set where $x_i \sim \mathcal{N}(\mu_k, \sigma_k^2)$,. Then we define K types of error patterns: for $k \in \{1, 2, \dots, K\}$, the k-th error pattern is

$$\begin{aligned} e_i^{(k)} &\triangleq \widehat{x}_{\text{W,GM},i} - x_i \\ &= \frac{\sigma_1^2 w_i + \sigma_u^2\mu_1}{\sigma_1^2+\sigma_u^2} - x_i \\ &\sim \mathcal{N}\left(\frac{\sigma_u^2}{\sigma_1^2+\sigma_u^2}\mu_1 - \frac{\sigma_u^2}{\sigma_1^2+\sigma_u^2}\mu_k, \frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_k^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2\right), \end{aligned}$$

where

$$i \in \mathcal{I}_k \triangleq \{i : x_i \sim \mathcal{N}(\mu_k, \sigma_k^2)\}.$$

Because the variances $\sigma_u^2, \sigma_1^2, \sigma_2^2, \dots, \sigma_K^2 > 0$ are constants, and $\sigma_1^2 = \max_{k\in\{1,2,\dots,K\}} \sigma_k^2$,

$$\frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_1^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2 = \max_{k\in\{1,2,\dots,K\}}\left(\frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_k^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2\right), \tag{E.1}$$

which shows that the first error pattern $e_i^{(1)}$ has the greatest variance.

Maximum of error patterns: Define the set A_ϵ as

$$A_\epsilon \triangleq \left\{\mathbf{x} : \left|\frac{|\mathcal{I}_1|}{N} - s_1\right| < \epsilon_1, \left|\frac{|\mathcal{I}_2|}{N} - s_2\right| < \epsilon_2, \dots, \left|\frac{|\mathcal{I}_K|}{N} - s_K\right| < \epsilon_K\right\},$$

where $\sum_{k=1}^{K} |\mathcal{I}_k| = N$, and $\epsilon_k \to 0^+$ as a function of N for $k \in \{1, 2, \ldots, K\}$. Applying a similar derivation to that of (D.14), we obtain that

$$\lim_{N\to\infty} \Pr\left(\frac{\max_{i\in\mathcal{I}_1} |\widehat{x}_{\text{W,GM},i} - x_i|}{\max_{j\in\mathcal{I}_k} |\widehat{x}_{\text{W,GM},j} - x_j|} > \frac{1-\Delta}{(1+\Delta)^2} \cdot \frac{\sqrt{\frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_1^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2}}{\sqrt{\frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_k^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2}} \middle| \mathbf{x} \in A_\epsilon \right) \tag{E.2}$$

$$= \lim_{N\to\infty} \Pr\left(\frac{\max_{i\in\mathcal{I}_1} |\widehat{x}_{\text{W,GM},i} - x_i|}{\max_{j\in\mathcal{I}_k} |\widehat{x}_{\text{W,GM},j} - x_j|} \geq 1 \middle| \mathbf{x} \in A_\epsilon \right) \tag{E.3}$$

$$= 1,$$

for any $k \neq 1$. Equation (E.3) is valid because (E.2) holds for any $\Delta > 0$, and $\frac{\sqrt{\frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_1^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2}}{\sqrt{\frac{\sigma_u^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_k^2 + \frac{\sigma_1^4}{(\sigma_1^2+\sigma_u^2)^2}\sigma_u^2}} \geq 1$ is derived from (E.1).

Hence,

$$\lim_{N\to\infty} E\left[\frac{\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{W,GM}}\|_\infty}{\sqrt{\ln(N)}} \middle| \mathbf{x} \in A_\epsilon \right] \lim_{N\to\infty} E\left[\frac{\max_{i\in\mathcal{I}_1} |x_i - \widehat{x}_{\text{W,GM},i}|}{\sqrt{\ln(N)}} \middle| \mathbf{x} \in A_\epsilon \right]. \tag{E.4}$$

Equation (E.4) shows that the maximum absolute error of the Wiener filter relates to the Gaussian component that has the greatest variance.

Optimality of the Wiener filter: Then applying similar derivations of equations (D.18) and (D.19),

$$\lim_{N\to\infty} \left(E\left[\frac{\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty}{\sqrt{\ln(N)}} \middle| \mathbf{x} \in A_\epsilon \right] - E\left[\frac{\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{W,GM}}\|_\infty}{\sqrt{\ln(N)}} \middle| \mathbf{x} \in A_\epsilon \right] \right) \geq 0.$$

Typical set: Similar to the derivation of (D.24), we obtain the probability of $\mathbf{x} \in A_\epsilon$ ([28], page 59),

$$\Pr(\mathbf{x} \in A_\epsilon) > 1 - \delta,$$

where

$$\delta = \sum_{k=1}^{K} \epsilon_k \left|\log_2(s_k)\right|.$$

Finally,

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\text{W,GM}}\|_\infty\right]}{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} < 1 + \delta,$$

where $\delta \to 0^+$, and thus

$$\lim_{N\to\infty} \frac{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\mathrm{W,GM}}\|_\infty\right]}{E\left[\|\mathbf{x} - \widehat{\mathbf{x}}_{\ell_\infty}\|_\infty\right]} = 1.$$

Appendix F

Proof of Lemma 1

It has been shown [12, 43] that for an i.i.d. standard Gaussian sequence $\widetilde{g}_i \sim \mathcal{N}(0,1)$, where $i \in \{1,2,\ldots,N\}$, the maximum of the sequence, $\max_i \widetilde{g}_i$, converges to $\sqrt{2\ln(N)}$ in probability, i.e.,

$$\lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N}\widetilde{g}_i}{\sqrt{2\cdot\ln(N)}}-1\right| < \Delta\right) = 1,$$

for any $\Delta > 0$. Therefore, for an i.i.d. non-standard Gaussian sequence $g_i \sim \mathcal{N}(\mu,\sigma^2)$, $\frac{g_i-\mu}{|\sigma|} \sim \mathcal{N}(0,1)$, and it follows that

$$\lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N}(g_i-\mu)}{\sqrt{2\sigma^2\cdot\ln(N)}}-1\right| < \Delta\right) = 1, \tag{F.1}$$

for any $\Delta > 0$. We observe that, for a given μ, the following probability equals 1 for sufficient large N, and therefore,

$$\lim_{N\to\infty} \Pr\left(\left|\frac{-\mu}{\sqrt{2\sigma^2\cdot\ln(N)}}-0\right| < \Delta\right) = 1, \tag{F.2}$$

for any $\Delta > 0$. Combining (F.1) and (F.2),

$$\lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N} g_i}{\sqrt{2\sigma^2\cdot\ln(N)}}-1\right| < 2\Delta\right) = 1,$$

for any $\Delta > 0$, which owing to arbitrariness of Δ yields

$$\lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N} g_i}{\sqrt{2\sigma^2\cdot\ln(N)}}-1\right| < \Delta\right) = 1. \tag{F.3}$$

Equation (F.3) suggests that, for a sequence of i.i.d. Gaussian random variables, $g_i \sim \mathcal{N}(\mu, \sigma^2)$, the maximum of the sequence is not affected by the value of μ.

On the other hand, the i.i.d. Gaussian sequence $(-g_i) \sim \mathcal{N}(-\mu, \sigma^2)$ satisfies

$$\lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N}(-g_i)}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta\right) = 1.$$

Hence,

$$\begin{aligned}
& \lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N}|g_i|}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta\right) \\
= & \lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N} g_i}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta \text{ and } \left|\frac{\max_{1\le i\le N}(-g_i)}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta\right) \\
= & \lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N} g_i}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta\right) - \lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N} g_i}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta \text{ and}\right. \\
& \left.\left|\frac{\max_{1\le i\le N}(-g_i)}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| > \Delta\right) \\
= & \lim_{N\to\infty} \Pr\left(\left|\frac{\max_{1\le i\le N} g_i}{\sqrt{2\sigma^2 \cdot \ln(N)}} - 1\right| < \Delta\right) - 0 \\
= & 1,
\end{aligned}$$

for any $\Delta > 0$.