TECHNISCHE UNIVERSITÄT WIEN

INSTITUTE OF TELECOMMUNICATIONS

MASTER THESIS

Compressed Sensing for Graph Signals

By

Ahmed Tawfik

(1131618)

Advisor:

Univ.Prof. Dipl.-Ing. Dr.-Ing.

Norbert Görtz

March 2018

Abstract

In this thesis we are going to tackle the problem of estimating a sparse graph signal with an unknown frequency support set of known size K from a sampled noisy version. Not knowing the location of the non-zero Fourier coefficients and by taking a number M of samples larger than K will give us a compressed sensing problem. The Bayesian Approximate Message Passing (BAMP) algorithm is used to solve the compressed sensing problem and recover the original signal from few coefficients.

Key words: Graph Signal, Unknown Frequency Support Set, Compressed sensing, BAMP algorithm.

Acknowledgements

I would first like to thank my thesis advisor Prof. Norbert Görtz of the Telecommunication department at TU Wien. The door to Prof. Görtz office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently gave me elaborative and informative explanation for my questions which gave me a solid understanding of the thesis topic. Overall, I felt encouraged by his positivity, driven by his high standards, and inspired by his work ethic.

I would also like to thank all my friends and colleagues in Austria for the encouragement and motivational conversations we had together and putting up to my stressed mood during stressful times.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Table of Contents

Chapter 1

Introduction

In the recent years, we are witnessing a huge increase in the amount of data being transferred and the networks are becoming bigger and more complex. Using the classical signal processing operations to process and represent this huge amount of data and complex signals is not efficient anymore. The goal is to find a feasible way to represent these complex signals and to be able compress the data as much as possible, so after transmission they can be perfectly reconstructed.

Graph signals can represent complex structured signals and still have the ability to carry out all the classical signal processing techniques like sampling, Fourier transform and others. In this thesis we are going to tackle the problem of estimating a sparse graph signal with an unknown frequency support set from a sampled noisy version. Not knowing the location of the K non-zero Fourier coefficients and by taking a number M of samples larger than K will give us a compressed sensing problem.

1.1 Summary of the Results

Compressed sensing gives the ability to reconstruct a signal from just few coefficients. The BAMP algorithm can be used to solve the compressed sensing problem and iteratively estimate the graph signal where the noise variance is unknown. The sensing matrix used is generated by the singular value decomposition of the weight matrix (SVD matrix). After many simulations of different scenarios we found out (as expected) that the

more samples M used, the better the quality of reconstruction. The quality of reconstruction improved significantly starting from $M \geq 2K$ and this is due to elementary null-space considerations of the compressed sensing problem which can also be proved algebraically. Also this is considered a practical advantage, where the graph signal dimension N is relatively high, yet M can remain the same as used for a lower dimension as long as it satisfies this condition.

To check the loss in the reconstruction process produced using the SVD matrix, the Gaussian matrix which is considered the ideal case was used. Also the Coherence property of these two matrices was calculated and were almost the same.

1.2 Thesis Overview

The following thesis is divided to five chapters. Chapter 2 introduces graph signals and gives a brief overview on this topic. It starts by defining a main component of graph signals which is the weight matrix and the eigen system decomposition of it. Also in this chapter the sampling support set and the frequency support set are defined and how a graph signal can be sampled. Finally we state the theorem that shows the condition for perfect recovery of a sampled graph signal.

Chapter 3 reviews compressed sensing; the chapter starts by setting the compressed sensing problem and mentioning every component in the problem. Then it focuses on one important component which is the sensing matrix and that by discussing three important properties of it. Last part of the chapter mentions how the sensing problem can be viewed as a ℓ_0 minimization problem. In this thesis, we are interested in one type of recovery algorithm which is called probabilistic algorithms; it will be discussed more in Chapter 4.

Chapter 4 starts by mentioning an example of a probabilistic recovery algorithm called Approximate Message Passing (AMP). Then description is given of the problem setting with derivations depending on probabilistic techniques. Finally, a detailed description follows of one variant of AMP called BAMP. All the equations needed for this iterative algorithm are available in this chapter. The BAMP algorithm is used in our work for this thesis.

Chapter 5 gives a detailed description of the work done in this thesis using the theoretical background mentioned in the previous chapters. The chapter starts by introducing the main problem that we are tackling in this thesis and the settings configured to create a simulated scenario. A number of simulations were done to check the effect of using different matrix and graph signal dimensions, also under different signal to noise ratios. These results are presented and described in this chapter. The same simulations were repeated but for the Gaussian matrix as the sensing matrix and a comparison is given between the old and the new simulations. Finally, the Coherence property of the sensing matrices was calculated for different dimensions.

All the theoretical knowledge that was used in this thesis, its sources are mentioned in the Bibliography section. In Appendix A, the equations used for rescaling the sensing matrix can be found, while in Appendix B there are more simulated results that are described in Chapter 5.

Chapter 2

Fundamentals of Graph Signals and Graph Signal Recovery

Signal processing on graphs is rooted in classical discrete signal processing of signals, where the simple classical signal processing operation, the delay shift, can be applied on graphs where it is called a graph shift. This chapter will start with an overview on graph signals and their Fourier transform. Then it is shown how the values of the graph frequencies are represented and how they play an important role in defining the bandwidth of the graph signal. Moreover an important operation on graph signals which is sampling is explained, and a sampling matrix is to be compressed. Finally, the recovery of the sampled signal will be discussed and how to achieve perfect recovery, which depends on the graph signal and the interpolation operation. All this fundamental knowledge on graph signals and operations carried with them will be used for the setup up of the simulations for this thesis. This chapter is based on [1] [2] [3].

2.1 Graphs

A graph is a structure that represents a set of objects showing the connections and relations between them. Irregular and complex signals can be represented by graphs by using discrete signal processing on graphs.

In discrete signal processing a graph $G = (\mathcal{V}, \mathbf{W})$, where $\mathcal{V} = \{v_0, \dots, v_{N-1}\}$, is a set of nodes of the graph and $\mathbf{W}$ the graph shift $N \times N$ matrix, for the real elements $w_{ij} \in \mathbb{R}$; the graph shift characterizes the connections between the nodes. The graph G can be directed or undirected, and the relation between two nodes v_i and v_j does not have to be symmetric.

2.2 Graph signals

A graph signal represents the mapping of the signal coefficients, x_j, where $j = 1,2,\dots N$, to the graph nodes v_j as illustrated in Figure 2.1. The graph signal can be written as $x = (x_1, x_2, \dots, x_N)^T$ which is an N-dimensional column vector with real or complex signal components.

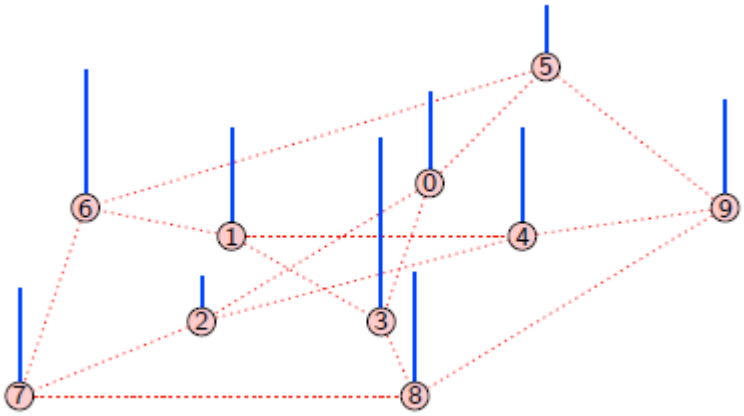


Figure 2.1 A Graph signal represented on a graph with 10 nodes (source: Antonio G. Marquesy, Santiago Segarraz, Alejandro Ribeiro, King Juan, Carlos University, Massachusetts Institute of Technology University of Pennsylvania, *Graph Signal Processing: Fundamentals and Applications to Diffusion Processes*, August 2016).

A graph filter can be seen as the generalization of time shift or as the basic filtering operation, like in classical signal processing, where the value at a node is replaced by the weighted sum of the values of its neighbor nodes: $x' = \mathbf{W}\, x$.

2.2.1 Graph Fourier transform

The Fourier transform [2] is the decomposition of the signal into basis elements, this basis is the eigenbasis of the graph shift **W** and is invariant to filtering. Let us assume that within the graph shift **W**, *N* linearly independent eigenvectors exist i.e **W** has a complete eigenbasis. **W** can be decomposed according to:

$$\mathbf{W} = \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V^{-1}} \tag{2.1}$$

The columns of matrix $\mathbf{V}$ are the eigenvectors of the matrix $\mathbf{W}$, and $\mathbf{\Lambda}$ is a diagonal matrix composed of the eigenvalues of $\mathbf{W}$.

For undirected graphs $\mathbf{W}$ has full rank, is real and symmetric and has complete orthonormal eigenbasis. In this case all the eigenvalues will have real values and for the decomposition of $\mathbf{W}$ as in (2.1), $\mathbf{V^{-1}} = \mathbf{V^{T}}$. For directed graphs with existing complete eigenbasis, the columns of matrix $\mathbf{V}$, which are linearly independent, can be used as a basis for the expansion of graph signal.

Hence, the graph Fourier transform of graph signal x is defined as:

$$\hat{x} = \mathbf{V^{-1}}\, x. \tag{2.2}$$

The vector $\hat{x}$ represents the graph Fourier coefficients which describes the frequency content of x, which will be discussed further later. The inverse graph Fourier transform is

$$x = \mathbf{V}\, \hat{x}\ . \tag{2.3}$$

2.3 Bandwidth of graph signal

Generally in sampling theory, the bandwidth of the signal is an important parameter for efficient sampling. For graph signals the potential frequency content is determined by the eigenvalues of $\mathbf{W}$ which represent the values of the graph frequencies.

Every coefficient in $\hat{x}$ is the strength of a graph frequency component, so in [1] the bandwidth of a graph signal is defined by the number of non-zero coefficients in $\hat{x}$. By doing so, the sampling framework in [1] gives the opportunity to use simple tools from linear algebra for sampling of signal with complicated (graph) structure.

2.3.1 Band-limited graph signals

For a perfect graph signal recovery, an important class of graph signals has to be stated and that is *bandlimited graph signals*. As in [2], the term *frequency support set* with size K is defined as

$$\mathcal{K} = \{\mathcal{K}_1, \mathcal{K}_2, \ldots., \mathcal{K}_K\} \tag{2.4}$$

With $|\mathcal{K}| = K$ and

$$\mathcal{K}_i \in \{1,2, \ldots., N\} \text{ and } \mathcal{K}_i \neq \mathcal{K}_j \text{ when } i \neq j,$$

so that for the coefficients of $\hat{\mathrm{x}}_k$, the Fourier-transform of the graph signal x, we have

$$\hat{\mathrm{x}}_k = 0 \text{ for all } k \in \{1,2, \ldots., N\} \setminus \mathcal{K} \tag{2.5}$$

A graph signal is *bandlimited* if it can be composed by number of eigenvectors fewer $(K < N)$ than the number of the eigenvectors existing in the columns of matrix $\mathbf{V}$, while a graph signal that is not bandlimited is called *full-band graph signal*. Such K is called the *bandwidth* of x.

As we just mentioned that $\hat{\mathrm{x}}_k \neq 0$ only for $k \in \mathcal{K}$, as stated in [2] the non-zero Fourier-coefficients can be computed according to

$$\hat{\mathrm{x}}_{\mathcal{K}} = (\mathbf{V^{-1}}\,\mathrm{x})_{\mathcal{K}} = (\mathbf{V^{-1}})_{(\mathcal{K},:)}\mathrm{x}, \tag{2.6}$$

The notation $(\mathbf{C})_{(\mathcal{K},:)}$ means that the rows according to the subset $\mathcal{K}$ are chosen and ":" for all the columns of matrix $\mathbf{C}$. By inverting the Fourier transform we get

$$\tilde{\mathrm{x}} = \mathbf{V}_{(\mathcal{K},:)}\,\hat{\mathrm{x}}_{\mathcal{K}} = \mathbf{V}_{(\mathcal{K},:)}\,(\mathbf{V^{-1}})_{(\mathcal{K},:)}\mathrm{x}, \tag{2.7}$$

It can be observed that $\tilde{\mathrm{x}} = \mathrm{x}$ when x is bandlimited as shown in equation (2.7).

2.4 Sampling

A graph signal x can be sampled by taking $M(M < N)$ nodes of x to produce a sampled signal $\mathrm{x}_{\mathcal{M}}$ where $\mathcal{M} = (\mathcal{M}_0, \dots, \mathcal{M}_{M-1})$ represent the sequence of the sampled indices, and the set $\mathcal{M}$ named the *sampling support set*, so that $\mathcal{M}_i \in \{0,1, \dots, N-1\}$. A sampling matrix $(M \times N)$ [2] can be defined as

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if } j = \mathcal{M}_i \\ 0 & \text{otherwise} \end{cases} \quad \text{for} \quad \begin{matrix} i = 1,2,\dots,M \\ j = 1,2,\dots,N \end{matrix} \tag{2.8}$$

where $|\mathcal{M}| = M, \quad \mathcal{M}_i \neq \mathcal{M}_l \quad \text{for} \quad i \neq l.$

Thus the sampling matrix relates the sampled graph signal to the unsampled graph signal as follows,

$$\mathrm{x}_{\mathcal{M}} = \mathbf{A}\mathrm{x} \tag{2.9}$$

There are two strategies of sampling [1]: *random sampling* in which the sample nodes are randomly chosen; and *experimentally designed* sampling where the samples are chosen before the experiment is conducted. After defining the sampling support, it can be stated that perfect recovery of a graph signal is possible, when $\hat{\mathrm{x}}$ has K non-zero components and K must not be larger than number of samples taken for the graph signal M.

2.5 Sampled graph signal recovery

As we just mentioned how a graph signal x can be sampled by multiplying it by a sampling matrix **A**, this brings up to the opposite case of which is interpolation. As stated in [1], an interpolation operator **Φ** can be defined such that

$$\mathrm{x}' = \mathbf{\Phi}\, \mathrm{x}_{\mathcal{M}} \tag{2.10}$$

Where $\mathrm{x}' \in \mathbb{R}^N$ and it recovers x from $\mathrm{x}_{\mathcal{M}}$.

The recovery of the graph signal depends on the graph signal itself and the sampling matrix used. For perfect recovery some conditions must be fulfilled and that will be shown in the following theorem:

Theorem 1 [4.2]:

Let the $M \times N$ sampling matrix **A** *satisfy the following:*

$$rank\left(\mathbf{A}\,\mathbf{V}_{(:,\mathcal{K})}\right) = K, \tag{2.11}$$

where $\mathbf{V}_{(:,\mathcal{K})}$ *is a $N \times K$ matrix with an index set $\mathcal{K}$ to address K linear independent vectors of dimension N in the columns of* $\mathbf{V}$. *For a graph signal x that is bandlimited to the frequency support set $\mathcal{K}$ of size* K, we have

$$\mathrm{x} = \tilde{\mathrm{x}} = \mathbf{\Phi}\,\mathrm{x}_{\mathcal{M}} = \mathbf{\Phi}\mathbf{A}\mathrm{x} \tag{2.12}$$

and perfect recovery is achieved by choosing

$$\mathbf{\Phi} = \mathbf{V}_{(:,\mathcal{K})}\mathbf{U} \tag{2.13}$$

with the $K \times M$ matrix **U** *such that*

$$\mathbf{U}\mathbf{A}\mathbf{V}_{(:,\mathcal{K})} = \mathbf{I} \tag{2.14}$$

with **I** the *$K \times K$ identity matrix.*

By checking Theorem 1 for the case $M > K$, the matrix $\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}$ will not be a square matrix. So the interpolation operator $\mathbf{\Phi}$ will be defined as the pseudoinverse of the $\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}$**:**

$$\mathbf{U} = \left(\left(\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}\right)^{\boldsymbol{H}}\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}\right)^{-\mathbf{1}}\left(\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}\right)^{\boldsymbol{H}} \tag{2.15}$$

For $M = K$, the matrix $\mathbf{\Phi}$ will just be defined as the inversion of the square matrix $\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}$. $\mathbf{A}\mathbf{V}_{(:,\mathcal{K})}$ can be simplified to $\mathbf{V}_{(\mathcal{M},\mathcal{K})}$, as the sampling matrix, which is defined by the sampling support set $\mathcal{M}$, only picks rows from $\mathbf{V}_{(:,\mathcal{K})}$. After this simplification the interpolation formula can be written as:

$$\tilde{\mathrm{x}} = \mathbf{V}_{(:,\mathcal{K})}\left(\left(\mathbf{V}_{(\mathcal{M},\mathcal{K})}\right)^{\boldsymbol{H}}\mathbf{V}_{(\mathcal{M},\mathcal{K})}\right)^{-\mathbf{1}}\left(\mathbf{V}_{(\mathcal{M},\mathcal{K})}\right)^{\boldsymbol{H}}\mathrm{x}_{\mathcal{M}}. \tag{2.16}$$

Graphs can represent irregular and complex signals while the graph shift represents the relation between the graphs nodes. By the end of this chapter it is now clear how graph signals represent the mapping of signal values to the graph nodes and how they are represented by appropriate vectors. Moreover the Fourier transform of the graph signal was

discussed. Important operations on graph signals like sampling and interpolation were explained, as they take an important role in the simulations in this thesis. Finally we have seen that for perfect recovery of a sampled graph signal using a sampling matrix with $M > K$ is alone not sufficient the operator also has to fulfil the condition mentioned in (2.11).

Chapter 3

Fundamentals of Compressed Sensing and Signal Recovery

Nyquist's and Shannon's studies had a huge impact on the communication world. They showed that for a continuous-time signal of finite bandwidth perfect recovery is possible from *sampling* it at a frequency at least twice the highest frequency present in that signal. With the enormous increase in the amount of data used nowadays, it became very hard and complex to detect such a large amount of samples. This created the motivation of finding different representations for the signal in order to sample it using fewer samples. There is the sparse representation, i.e. the signal is defined by a much smaller number of non-zero coefficients than the length of the signal, and the compressible representation, i.e. signal approximated by a signal with much fewer non zero coefficients relative to its length.

Compressed sensing gives the ability to represent many signals using a few non-zero coefficients which can be later recovered. It needs fewer measurements to sense the signal (i.e. sparse signals) compared to the Shannon sampling theorem. The measurements needed to be linear and must not depend on each other. Let's define a compressed sensing system, i.e. an underdetermined system, which takes m linear measurements for a signal vector x $\in$ $\mathbb{R}^n$ ($m \ll n$) where:

$$y = A\mathrm{x}\,, \tag{3.1}$$

$y \in \mathbb{R}^m$ is the observed vector and A is the $m \times n$ sensing matrix see illustration in Figure 3.1. This chapter will address two important points. The first one is the construction and properties of matrix A and the second point is the reconstruction of the signal x.

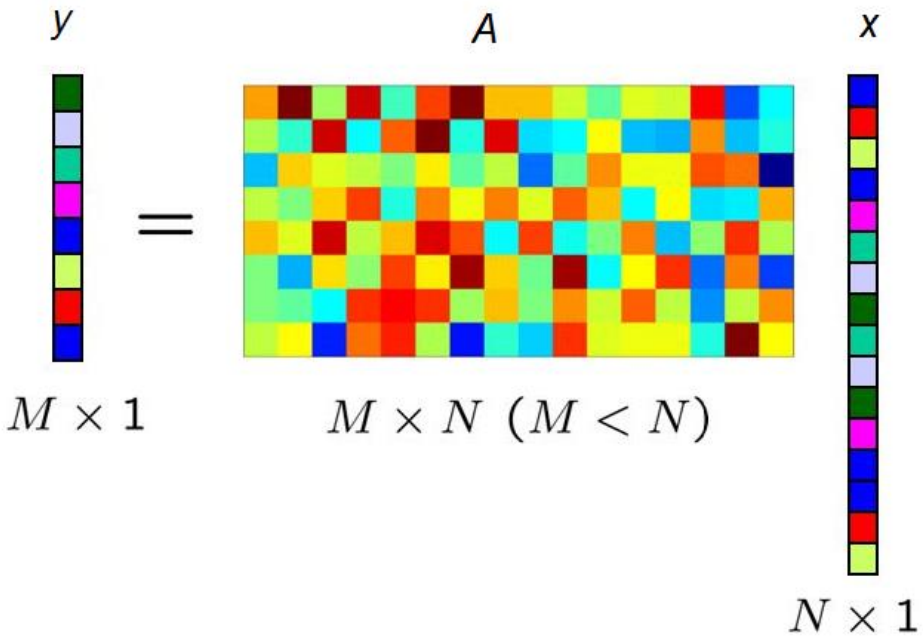


Figure 3.1 Representation of the effect of the sensing matrix A on the signal x. (Source: https://www.quora.com/topic/Compressed-Sensing)

3.1 Sensing matrix

The sensing matrix A is non-adaptive i.e. fixed and does not depend on the signal x. It can be perceived as a mapper that maps the signal from $\mathbb{R}^n$ to $\mathbb{R}^m$, thus it condenses the information in a signal into a small amount of data as illustrated in Figure 3.1. From equation (3.1) signal recovery can be modeled as an ℓ_0 minimization problem [4]

$$\text{minimize} \quad \|\mathrm{x}\|_0 \qquad (3.2)$$

$$\text{subject to} \quad A\mathrm{x} = \mathrm{y},$$

where $\|\mathrm{x}\|_0$ is the number of non-zero elements in vector x. The minimization of the ℓ_0-"norm" is a non-convex problem so it is simpler to solve the related problem that results from relaxing it to the convex ℓ_1 minimization problem:

$$\text{minimize} \quad \|\mathrm{x}\|_1 \qquad (3.3)$$

$$\text{subject to} \quad A\mathrm{x} = \mathrm{y}.$$

The optimal solution for the ℓ_0 minimization is very close to the solution of the ℓ_1 minimization but the matrix A has to have certain conditions. These conditions should be taken into consideration when designing the matrix A.

3.1.1 Null space conditions (NSC)

A formal definition of null space of matrix A [5]:

$$\mathcal{N}(A) = \{z : Az = 0\} \tag{3.4}$$

From the work of Gribonval and Nielsen they gave the following theorem:

Theorem 2: [6]

Given a matrix $A \in \mathbb{R}^{n\times N}$, *every* s*-sparse vector* $x \in \mathbb{R}^N$ *is the unique solution of the* ℓ_1*minimization with* $y = Ax$ *if and only if A satisfies the null space property of order* s.

In other words the matrix A satisfies the NSC [4] for a positive integer k if:

$$\|z_K\|_1 < \|z_{\bar{K}}\|_1, \tag{3.5}$$

holds true for all $z \in \{z|\ Az = 0, z \neq 0\}$ and for all subsets of $K \subseteq \{1,2, \dots n\}$ where K is an index set, z_K is the part of the vector z over the index set K and $\bar{K}$ is the complement of K. In [7] it is shown that the NCS is related to a proportion diameter α_k which is defined as:

$$\alpha_k = \max_{\{z:Az=0, z\neq 0\}} \max_{\{K:|K|\leq k\}} \frac{\|z_K\|_1}{\|z\|_1} \tag{3.6}$$

α_k can be used to verify the NSC and there is some work on algorithms to calculate it. Calculating α_k is complex, which renders NCS not used so much practically. Also it does not account for signals contaminated with noise, which makes us consider stronger conditions.

3.1.2 Restricted isometry property (RIP)

Candes and Tao [8] introduced the RIP and showed how it is crucial for the optimal recovery of a signal using compressed sensing. For a sparse vector x, there is a sensing matrix A that satisfies the RIP of order $2k$ such that:

$$(1-\delta_k)\|A\mathrm{x}\|_2^2 \leq \|A\mathrm{x}\|_2^2 \leq (1+\delta_k)\|A\mathrm{x}\|_2^2, \tag{3.7}$$

where δ_k is the RIP constant ($0 < \delta_k < 1$). This shows that, if matrix A satisfies such a criterion, the distance between any two k sparse vectors is preserved such that they can be recovered as two different vectors (and not as one). In (3.7) the bounds are symmetric around 1 which is not the case in practice. In practice (3.7) becomes:

$$\alpha\|\mathrm{x}\|_2^2 \leq \|A\mathrm{x}\|_2^2 \leq \beta\|\mathrm{x}\|_2^2 \tag{3.8}$$

where $0 < \alpha \leq \beta < \infty$ but it is always possible to scale A back so it can satisfy the boundaries in (3.7).

For achieving the RIP there is a bound for the number of measurements needed, and according to a theorem from [5] this bound does not depend on the RIP constant δ_k and only depends on the dimensions of the problem. If we have matrix A with dimension $M \times N$ and it satisfies the RIP of order $2k$, then we would have the following:

$$M \geq Ck \log\left(\frac{N}{k}\right), \tag{3.9}$$

where $C = 1/2\log(\sqrt{24}+1) \approx 0.28$ and $\delta < \frac{1}{2}$.

Finally it is shown that according to a theorem from [4] that if a matrix satisfies the RIP, then it also satisfies the NSC, although the RIP is strictly stronger than the NSC.

3.1.3 Mutual Incoherence Property (MIP)

The MIP is easier to calculate compared to NSC and RIP, therefore it is considered in a lot of practical work. The coherence of matrix A, defined as

$$\mu(A) = \max_{1\leq i<j\leq N} \frac{|\langle a_i, a_j\rangle|}{\|a_i\|_2 \|a_j\|_2} \tag{3.10}$$

where a_i represents column i in matrix A and the coherence has the following boundaries $\mu(A) \in \left[\sqrt{\frac{n-m}{m(n-1)}}, 1\right]$, where the lower bound is called the Welch bound [4]. So the coherence of a matrix can be interpreted as the largest inner-product (normalized) between any two different columns.

For a matrix A to obey the MIP [9] it must satisfy the following:

$$\mu(A) \leq B_0 \cdot (\log N)^{-1}, \tag{3.11}$$

where B_0 is some positive numerical constant, while for the case of the Gaussian matrix (i.e. with i.i.d. entries) $\mu(A) = \sqrt{(2 \log N)/M}$. Also in [10] it is stated that for exact recovery of a k-sparse signal, the following condition has to be satisfied:

$$(2k - 1)\mu(A) < 1 \tag{3.12}$$

If the matrix A satisfies (3.12), exact recovery is possible in the noiseless case and near optimal in the case of small noise that contaminates signal.

So it can be concluded for the matrix A in our signal setting (3.1), that the smaller the coherence value the better the recovery of the signal will be.

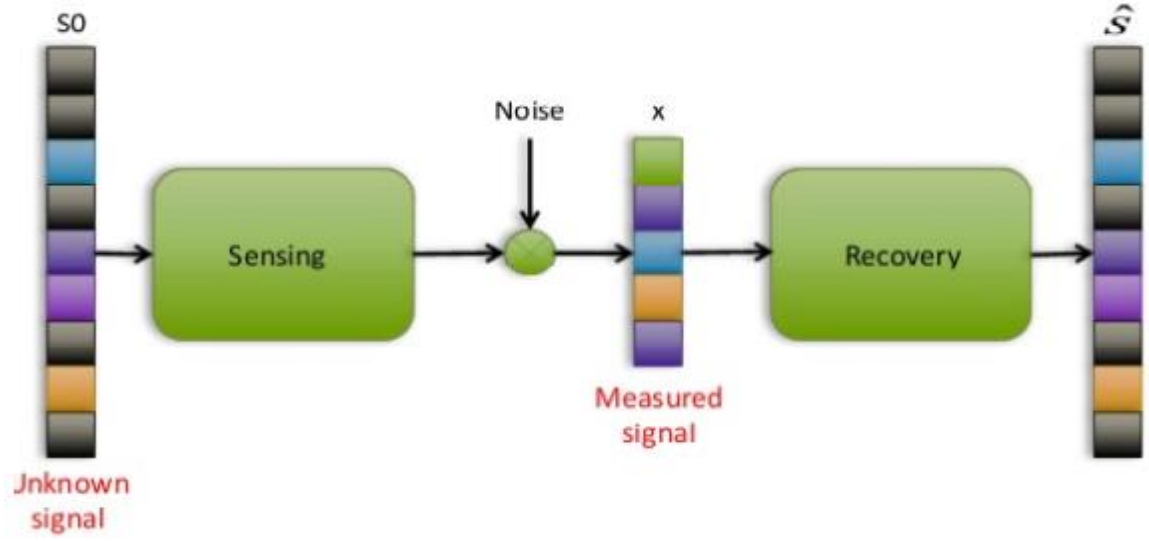


Figure 3.2 The overall process of solving a compressed sensing problem.

Source: Stefani Thomas, *Compressive Sensing Basics-Medical Imaging-MRI,* www.slideshare.net/thomasstefani169/compressive-sensing-basics-medical-imaging-mri, 2014

3.2 Signal recovery

The overall picture of the compressed system is illustrated in Figure 3.2. In the last part we discussed the sensing part and how the properties of the sensing matrix should be. In this part we will give an introduction to recovery and in the next chapter we will mention in detail the recovery algorithm used in this thesis.

As mentioned in the last part, the sensing problem can be viewed as a ℓ_0 minimization problem, and that was shown for the specific case of noiseless signals in (3.2). Given measurements y and a sparse signal x, a more generalized form of the optimization problem can be formulated as

$$\hat{\mathrm{x}} = \arg\min_{z} \|z\|_0 \quad \text{Subject to} \quad z \in \boldsymbol{\mathcal{B}}(\mathrm{y}), \tag{3.13}$$

where $\boldsymbol{\mathcal{B}}(\mathrm{y})$ is the cost function that ensures the consistency between the recovered signal and the measurements made. For example as in [5], in the case where there is no noise we have $\boldsymbol{\mathcal{B}}(\mathrm{y}) = \{z : \|A\mathrm{z} = y\|_2\}$, while with the presence of bounded noise $\boldsymbol{\mathcal{B}}(\mathrm{y}) = \{z : \|A\mathrm{z} - y\|_2 \leq \epsilon\}$.

There are different types of recovery algorithms, for example the convex optimization techniques like the ℓ_1 minimization algorithm, in which the non-convex ℓ_0 minimization problem is relaxed to a convex ℓ_1 minimization problem, or the greedy/iterative methods for solving these problems by iteratively getting an estimate of the signal till a convergence criterion is met. There is also the probabilistic approach, where the recovery is being handled as an estimation problem. Approximate Message Passing (AMP) or Bayesian Approximate Message Passing (BAMP) are examples of this approach. In this thesis BAMP was used and it will be discussed in more detail in the next chapter.

Modern transform coders would acquire the full signal, encode some coefficients and then discard the rest, which results in a huge waste of resources and power. In compressed sensing there is no need to this. It makes it possible to acquire data that are already compressed (by compressed sampling via A). Compressed sensing helps to recover sparse signals from just a few coefficients (relative to the dimension of the signal).

In this chapter we stated the compressed sensing problem, the importance of the sensing matrix in this system and the properties they should have to achieve good recovery of the signal.

Chapter 4

Bayesian Approximate Message Passing applied for Recovery of Sampled Graph Signals

There have recently been many studies on CS signal reconstruction algorithms. Some of these algorithms were mentioned in the last chapter, but in this chapter we are interested in Approximate Message Passing (AMP) algorithms. Message passing for compressed sensing recovery was first introduced by Sarvotham et al in [11] and then, recently, the AMP algorithm was proposed in [13]. The AMP algorithm is interesting for its low complexity and fast convergence as an iterative algorithm. As being described in [12] "*AMP algorithm applies the central limit theorem to sum-product belief propagation (BP) (or quadratic approximation to max-sum BP) followed by Taylor expansion to simplify the messages passing between nodes"*. The Onsager Correction term is what distinguishes the AMP algorithm from other iterative thresholding algorithms; this term changes the statistical properties of the reconstruction and it gives the ability for analyzing the reconstruction by a technique called State Evolution. State Evolution allows to predict the performance of the AMP algorithms and tune them for the best performance possible [13].

A variation of the AMP algorithm is the Bayesian Approximate Message Passing (BAMP) algorithm. BAMP exploits the prior knowledge of the signal. Although BAMP gives an approximate estimate of the signal, it is considered a very accurate and efficient algorithm. In this thesis the BAMP algorithm was used for recovering the signal, so before taking a deeper look at the algorithm used, let's set the problem on which the BAMP algorithm will be applied on. The problem setting is based on [14].

4.1 Problem Setting

Linear measurements are taken for an n-dimensional signal vector $\mathrm{x} = \{x_j, j = 1, \dots, n\}$ and with a sensing matrix **A** of dimensions $m \times n (m < n)$ producing observations arranged in vector y of dimension m,

$$\mathrm{y} = \mathbf{A}\mathrm{x} + \mathrm{w} \tag{4.1}$$

The matrix **A** is assumed to have its full possible rank m and its components are independently drawn realizations of a real random variable. The columns of **A,** which can be represented as vectors $\mathbf{A}_j, j = 1, \dots, n$, are assumed to have zero mean and be normalized to unit l_2-norm such that:

$$\mathbf{A} = \{\mathbf{A_1}, \mathbf{A_2}, \dots, \mathbf{A_n}\} \tag{4.2}$$
$$\text{with } \|\mathbf{A}_j\|_2 = 1 \quad \forall\, j$$

The signal vector y is contaminated by Gaussian noise which can be seen in (4.1) as an m-dimensional vector $\mathrm{w} = \{w_j, j = 1, \dots, m\}$ that is added. The components of vector w are assumed to independent and identically distributed Gaussian with variance $\sigma^2 > 0$.

From (4.1) the measurements y and matrix **A** are given and the signal vector x is unknown. The number of measurements m is smaller than the number of the unknown signal components n making (4.1) an underdetermined problem.

4.1.1 Prior Knowledge

One way to solve such problem is by getting a good estimate of x by the help of exploiting the *prior knowledge* of x. As we are mainly dealing with sparse signals that can be considered an extra information known about x before estimation, also x can be known to have a large number of components that take the values ± 1. What we will take into consideration concerning the prior knowledge of x, is that the probability distribution $p_{\mathrm{X}}(\mathrm{x})$ is known. Assuming the n-dimensional signal is sparse with s non-zero coefficients where $s \ll n$, a probability density function can be defined as [14]:

$$p_{X_j}(x_j) = \epsilon\delta(x_j) + (1 - \epsilon) f_{X_j}(x_j) \qquad \forall\, j = 1, \dots, n \tag{4.3}$$

where ϵ is the probability of zero components ($\epsilon = (n-s)/n$) and it is centered at x = 0 by a delta function, also all components of x are assumed to be independent. Moreover, $f_{X_j}(x_j)$ is the pdf of the non-zero signal samples which is also assumed to be known.

4.2 Estimation Problem

As mentioned in the previous Chapter 3.2, a probabilistic (estimation) approach is one way for estimating sparse signals. An optimization criterion is defined such that the expectation of the mean squared error is minimized to give an estimation of the signal vector.

$$\hat{\mathrm{x}} = \arg\min_{\tilde{\mathrm{x}}} \mathbb{E}_{\mathrm{X,W}}\{\|\mathbf{X} - \tilde{\mathrm{x}}\|_2^2 \mid \mathbf{Y} = \mathrm{y}\} \tag{4.4}$$

The estimation is taken over two random components which are the unknown signal vector **X** and the noise vector W given the *deterministically* known measurement vector y. To solve (4.4) the partial derivatives of this equation are taken for the vector components $\hat{x}_j$ and are set to zero giving the following result:

$$\hat{\mathrm{x}} = \mathbb{E}_{\mathrm{X,W}}\{\mathbf{X} \mid \mathbf{Y} = \mathrm{y}\} \tag{4.5}$$

which will be solved to give:

$$\hat{\mathrm{x}} = \int_{\mathbb{R}^n} \mathrm{x}\, p_{\mathbf{X}|\mathbf{Y}}(\mathrm{x}|\mathrm{y}) d\mathrm{x} \tag{4.6}$$

And using Bayes rule i.e. $p_{\mathbf{X}|\mathbf{Y}}(\mathrm{x}|\mathrm{y}) = p_{\mathbf{Y}|\mathbf{X}}(\mathrm{y}|\mathrm{x}) p_{\mathbf{X}}(\mathrm{x}) / p_{\mathbf{Y}}(\mathrm{y})$:

$$\hat{\mathrm{x}} = \frac{1}{p_{\mathbf{Y}}(y)} \int_{\mathbb{R}^n} \mathrm{x}\, p_{\mathbf{Y}|\mathbf{X}}(\mathrm{y}|\mathrm{x}) p_{\mathbf{X}}(\mathrm{x})\, d\mathrm{x}, \tag{4.7}$$

$p_{\mathbf{X}}(\mathrm{x})$ is the signal prior, and since components of signal x are assumed independent and each with pdf $p_{\mathrm{X}_j}(x_j)$, the prior pdf can be written as:

$$p_{\mathbf{X}}(\mathrm{x}) = \prod_{j=1}^{n} p_{\mathrm{X}_j}(x_j). \tag{4.8}$$

The pdf $p_{\mathbf{Y}|\mathbf{X}}(\mathrm{y}|\mathrm{x})$ describes the noisy measurement process, where from (4.1) we can get:

$$w_k = y_k - (\mathbf{A}\mathrm{x})_k, \tag{4.9}$$

where w_k represents the noise at the k-th component. Assuming having independent Gaussian noise with variance σ^2, we obtain:

$$p_{\mathbf{Y}|\mathbf{X}}(\mathrm{y}|\mathrm{x}) = \prod_{k=1}^{m} \frac{1}{\sqrt{2\pi}\sigma}\, e^{-(y_k-(\mathbf{A}\mathrm{x})_k)^2/(2\,\sigma^2)} \tag{4.10}$$

Inserting (4.8) and (4.10) in (4.7), we will get the following expression for the estimated signal $\hat{\mathrm{x}}$:

$$\hat{\mathrm{x}} = \frac{1}{p_{\mathbf{Y}}(y)} \int_{\mathbb{R}^n} \mathrm{x} \prod_{k=1}^{m} \frac{1}{\sqrt{2\pi}\sigma}\, e^{-(y_k-(\mathbf{A}\mathrm{x})_k)^2/(2\,\sigma^2)} \prod_{j=1}^{n} p_{\mathrm{X}_j}(x_j) \tag{4.11}$$

Solving (4.11) requires high dimensional integration, which makes it practically infeasible. BAMP introduces an efficient and accurate approach to approximately compute this equation also for large dimension n.

4.3 Bayesian-Optimal Approximate Message Passing (BAMP)

Mainly, BAMP is an iterative algorithm that decomposes $p_{\mathbf{Y}|\mathbf{X}}(\mathrm{y}|\mathrm{x})$ into marginals for the single signal components x_j and to compute those marginals with believe propagation [14]. We will use the same problem setting we mentioned previously with the m – dimensional observation vector y, $m \times n$ sensing matrix **A,** variance of noise σ^2 and prior pdf $p_{\mathrm{X}_j}(x_j)$ are known.

4.3.1 BAMP I Algorithm [14]

Measurement noise variance σ^2 is known. It starts at $t = 0$ with the following intializations:

$$\hat{\mathbf{x}}^{\mathbf{0}} = \mathbf{0}_{n\times 1} \qquad \text{(signal vector; dimension } n > m\text{)} \tag{4.12}$$

$$\mathbf{z}^{\mathbf{0}} = \mathbf{y} \qquad \text{(dimension } m \times 1\text{)} \tag{4.13}$$

$$c^0 = \sigma^2 + \frac{1}{m}\left\|\mathbf{z^0}\right\|_2^2 \quad \text{(scalar)} \tag{4.14}$$

Then for iterations $t = 1,2, \ldots$:

$$\mathbf{u}^{t-1} = \{u_1^{t-1}, u_1^{t-2}, \ldots\ldots, u_n^{t-1}\}^T = \hat{\mathrm{x}}^{t-1} + \mathbf{A}^T\mathbf{z}^{t-1} \tag{4.15}$$

$$\hat{x}_j^t = F\left(u_j^{t-1};\ c^{t-1}\right), \qquad j = 1,2, \ldots\ldots, n \tag{4.16}$$

$$v_j^t = G\left(u_j^{t-1};\ c^{t-1}\right), \qquad j = 1,2, \ldots\ldots, n \tag{4.17}$$

$$q_j^{t-1} = F'\left(u_j^{t-1};\ c^{t-1}\right), \qquad j = 1,2, \ldots\ldots, n \tag{4.18}$$

$$\hat{\mathbf{x}}_j^t = \{\hat{x}_1^t, \hat{x}_2^t, \ldots\ldots, \hat{x}_n^t\}^T \tag{4.19}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\,\hat{\mathbf{x}}^t + \mathbf{z}^{t-1}\frac{1}{m}\sum_{j=1}^{n} q_j^{t-1} \tag{4.20}$$

$$c^t = \sigma^2 + \frac{1}{m}\sum_{j=1}^{n} v_j^{t-1} \tag{4.21}$$

With scalar operators:

$$F\left(u_j;\ c\right) = \mathbb{E}_{X_j}\left\{X_j | U_j = u_j\right\} \tag{4.22}$$

$$G\left(u_j;\ c\right) = \mathrm{Var}_{X_j}\left\{X_j | U_j = u_j\right\} \tag{4.23}$$

$$F'\left(u_j;\ c\right) = \frac{d}{du_j}F\left(u_j;\ c\right) \tag{4.24}$$

The subscript t represents the iteration index, while the variables $\mathbf{u}^{t-1}$ are the auxiliary variables with the components u_j^{t-1}. Also $\mathbf{z}^t$, v_j^t and q_j^{t-1} are auxiliary variables. c^t is the current estimate of the noise variance and the computation of the scalar operations will be shown later in this chapter.

4.3.2 BAMP II Algorithm

This algorithm is for an unknown measurement noise variance. It starts at $t = 0$ with the following initializations:

$$\hat{\mathbf{x}}^{\mathbf{0}} = \mathbf{0}_{n\times 1} \qquad \text{(signal vector; dimension } n > m\text{)} \tag{4.25}$$

$$\mathbf{z^0} = \mathbf{y} \qquad \text{(dimension } m \times 1\text{)} \tag{4.26}$$

$$c^0 = \frac{1}{m}\left\|\mathbf{z^0}\right\|_2^2 \qquad \text{(scalar)} \tag{4.27}$$

Then for iterations $t = 1,2, \ldots$:

$$\mathbf{u}^{t-1} = \hat{\mathrm{x}}^{t-1} + \mathbf{A}^T\mathbf{z}^{t-1} \tag{4.28}$$

$$\hat{\mathbf{x}}_j^t = F\left(u_j^{t-1};\ c^{t-1}\right), \qquad j = 1,2, \ldots\ldots, n \tag{4.29}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\,\hat{\mathbf{x}}^t + \mathbf{z}^{t-1}\frac{1}{m}\sum_{j=1}^{n} F'\left(u_j^{t-1};\ c^{t-1}\right) \tag{4.30}$$

$$c^t = \frac{1}{m}\ \|\mathbf{z}^t\|_2^2 \tag{4.31}$$

with $F\left(u_j^{t-1};\ c^{t-1}\right)$ and $F'\left(u_j^{t-1};\ c^{t-1}\right)$ defined as in (4.22) and (4.24).

The main difference between the two algorithms is how the noise variance c^t is computed. In BAMP II there is no need to know the measurement noise variance σ^2, as it is approximated by the ℓ_2-norm of the residual $\mathbf{z}^t$. According to [14], although in BAMP II σ^2 is not known, yet the performance of this algorithm is the same as that of BAMP I. The stopping criterion for this iterative algorithm is by done by checking, if the estimate $\hat{\mathbf{x}}^t$ is not changing anymore. This is implemented by stopping the iterations if:

$$\|\,\hat{\mathbf{x}}^t - \,\hat{\mathbf{x}}^{t-1}\|_2 < \varepsilon\|\,\hat{\mathbf{x}}^t\|_2 \tag{4.32}$$

where ε is chosen to be a small factor such as $\varepsilon = 10^{-4} \ldots . 10^{-6}$.

4.3.3 Computation of Scalar Operators

It has been shown by the central limit theorem that the noise model that applies to the signal components in the BAMP iterations is Gaussian for large *n*. In this part the final equations for the scalar operators $F\left(u_j;\ c\right)$, $F'\left(u_j;\ c\right)$ and $G\left(u_j;\ c\right)$ will be given without derivations. All the equations are from [4.4] and this is also where their derivations can be found.

As mentioned in part 4.1.1 the prior knowledge is being exploited to give a good estimate for the signal, also it is being assumed we are dealing with sparse signals. We are interested in the case with sparse Gaussian prior (zero mean).

The pdf of the sparse Gaussian prior is:

$$p_X(x;\gamma,\sigma) = \gamma\delta(x) + (1-\gamma)\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2\sigma^2}x^2} \tag{4.33}$$

with γ is the probability of zero value $(0 \le \gamma \le 1)$ and σ is the variance of the Gaussian distribution if no discrete zero-component is generated by the source. The component indices *j* will be dropped from *x* and from *u* for compact notation.

For the optimal MMSE estimator with a sparse Gaussian prior:

$$F(u;c) = u\frac{q}{q+1}\ \frac{1}{1+N(u,\gamma,c,q)} \tag{4.33}$$

Where $q = \sigma^2/c$ and $N(u,\gamma,c,q)$:

$$N(u,\gamma,c,q) = \frac{\gamma}{1-\gamma}\sqrt{q+1}e^{-\frac{u^2}{2c}\frac{q}{q+1}} \tag{4.34}$$

Thus from (4.33) and (4.34), as the noise variance increases $(c \to \infty)$, the output is muted $(F(u;c) \to 0)$ and as $c \to 0$ the measurement will represent the true value of *x* $(F(u;c) \to u)$.

Finally the variance of the optimal MMSE estimator in (4.33) is:

$$G(u;c) = c\frac{d}{du}F(u;c) = c\,M(u,\gamma,q) + m(u,\gamma,q)F^2(u;c) \tag{4.35}$$

with

$$M(u,\gamma,q) = \frac{q}{q+1}\ \frac{1}{1+m(u,\gamma,q)} \tag{4.36}$$

and

$$m(u,\gamma,q) = \frac{\gamma}{1-\gamma}\sqrt{q+1}e^{-\frac{u^2}{2c}\frac{q}{q+1}} \tag{4.37}$$

with $q = \sigma^2/c$.

Approximate Message passing algorithms with their low complexity relative to other algorithms, solve the problem of recovering the signals from compressed sensing

measurements. After several studies, simulations and analysis on AMP algorithms, it was shown that these algorithms exhibit fast convergence rate. BAMP, being a version of AMP where it exploits the prior knowledge of the signal, has shown very good performance in recovering the signals efficiently and accurately. This is why BAMP is used in our work for this thesis to recover the signals.

This chapter introduced the theoretical background concerning BAMP, which will be used in our work and simulations and this will be discussed in the following chapter.

Chapter 5

Simulations

In Chapter 2 the concept of graph signals was discussed, how they are sampled and the conditions required for reconstructing the signal from the samples. The theoretical background for graph signals that was discussed in Chapter 2 was mainly taken from [4.2]. It was shown that for perfect recovery of a sampled graph signal x, two conditions must be fulfilled.

The first condition is that the graph signal should be bandlimited, meaning that the number of non-zero components K in the frequency support must not be larger than the number of sampled components M in the sampling support. The second condition, which is concerned with the sampling matrix, is to fulfil the rank criterion stated in Theorem 1 (2.11). If these conditions are fulfilled, there are some already known methods to perfectly recover the signal, one of them discussed in [4.2]. All these methods rely on the fact that the sampling and the frequency support sets are *known.*

The main problem we are tackling in this thesis is, what if the frequency support set is *unknown* and how can we still perfectly recover the signal. In this chapter we will introduce how this problem is solved, how it becomes a compressed sensing problem, for which the BAMP algorithm can be used to recover the graph signal, and we will introduce some simulations that were done to see the effect of different parameters on the recovery process. But first let's define our problem according to the theoretical knowledge that was presented in the previous chapters.

5.1 Problem Setting

Following what was introduced in Chapter 2, we are given a sampled graph signal as:

$$\mathrm{x}_{\mathcal{M}} = \mathbf{A}\,\mathrm{x}, \tag{5.1}$$

where $\mathrm{x}_{\mathcal{M}}$ is the observed sampled signal and x is a N-dimensional graph signal that is being sampled by $\mathbf{A}$ $(M \times N)$, the sampling matrix. From (2.3) we get:

$$\mathrm{x}_{\mathcal{M}} = \mathbf{A}\,\mathbf{V}\,\hat{\mathrm{x}} \tag{5.2}$$

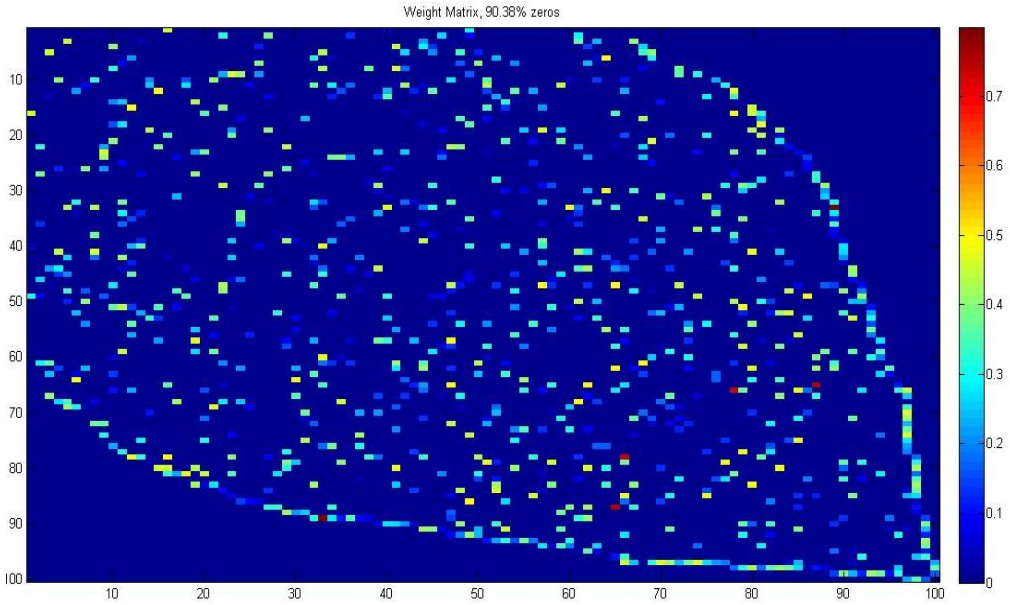
$\mathbf{V}$ $(N \times N)$ is the matrix which has in its columns the eigen vectors of the graph shift and $\hat{\mathrm{x}}$ contains the graph Fourier coefficients ordered in a $N \times 1$ vector as shown in (2.2). This would make the observed graph signal the following:

$$\mathrm{y} = \mathrm{x}_{\mathcal{M}} + w, \tag{5.3}$$

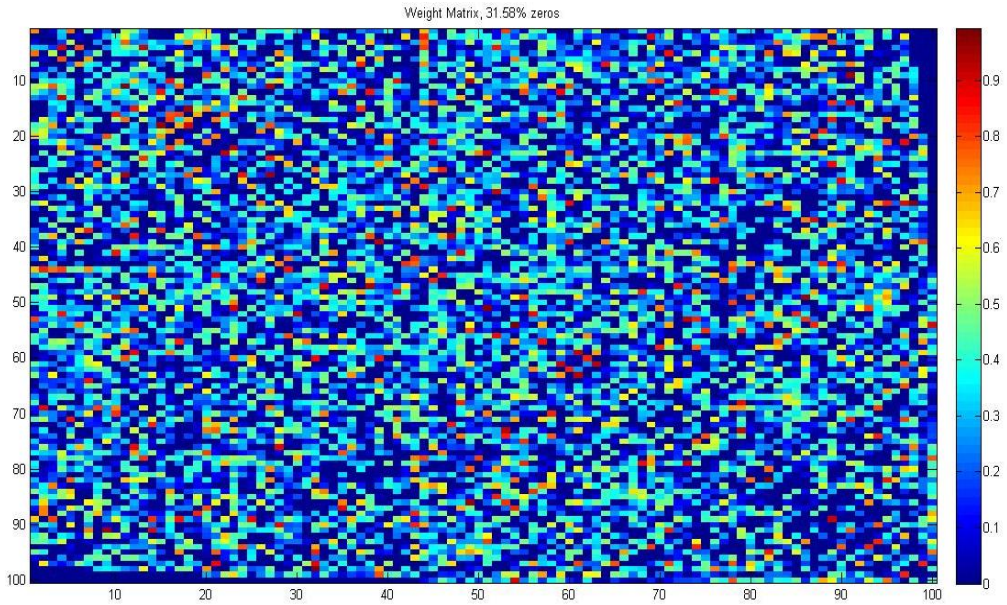
where the sampled graph signal is contaminated with random Gaussian noise. First, each parameter in (5.3) has to be defined according to the settings introduced in the following sections.

5.1.1 Weight Matrix

As mentioned in Chapter 2, the matrix $\mathbf{V}$ represents the eigen vectors of the weight matrix defined in (2.1). The weight matrix $\mathbf{W}$ is a $(N \times N)$ matrix. The entries of the $\mathbf{W}$ were chosen to be uniformly distributed pseudorandom numbers. Also $\mathbf{W}$ has a high sparsity level, as we are mainly dealing with sparse graph signals. In Figure 5.1 (a), we can get an idea of how the weight matrix was being represented and check the sparsity level from the percentage of the zero coefficients in the matrix. The nodes of the graph signal were implicitly renumbered such that the weight matrix would be as far as possible a diagonal matrix and that was realized by a standard Matlab function (symrcm()). Also it is being compared to another simulated matrix with low sparsity level as shown in Figure 5.1 (b).



(a)



(b)

Figure 5.1 (a) Weight matrix which will be used in the simulations with more than 90% of the entries are zeros (sparse matrix). (b) Weight matrix with almost 32% of the entries are zeros (for comparison).

5.1.2 Graph Fourier Vector

The vector $\hat{x}$ represents the graph Fourier coefficients which describe the frequency content of x. In our work we assumed that the prior knowledge of the signal is known, where the prior is sparse Gaussian with zero mean as given in (4.33). $\hat{x}$ will have K non-zero components, these components will be the Fourier transform of a Gaussian function. We will assume that the number of non-zero Fourier coefficients is known, but not the frequency support set.

5.1.3 Sampling Matrix

The sampling matrix **A** in (5.1) and (5.2) was designed to have the same properties as (2.8). By multiplication of the matrix **A** and **V**, (5.2) can be written as:

$$x_{\mathcal{M}} = \mathbf{B}\,\hat{x}, \tag{5.4}$$

where **B** is an $N \times N$ matrix representing the sensing matrix, making (5.3) a compressed sensing problem. In order to recover the signal, the same steps were followed as discussed in Chapter 4, where we start with (5.3) which is the same as (4.1). The only difference is that the matrix **B** has to be normalized to unit l_2-norm as in (4.2). This normalization will lead to some scaling of the signal prior (see Appendix A) which will directly affect the BAMP algorithm through the prior variance. The normalization is done using the equations from Appendix A.

5.1.4 Signal Recovery

The BAMP II algorithm [4.4] mentioned in Chapter 4 was used for signal recovery. The first simulation done in this thesis was to check the ability of the BAMP algorithm to recover the original signal in the graph signal recovery setting. Given the observation signal, the number of sampled indices M, the number of non-zero Fourier coefficients K and the sensing matrix **B**, an original graph signal was recovered as shown in Figure 5.2.

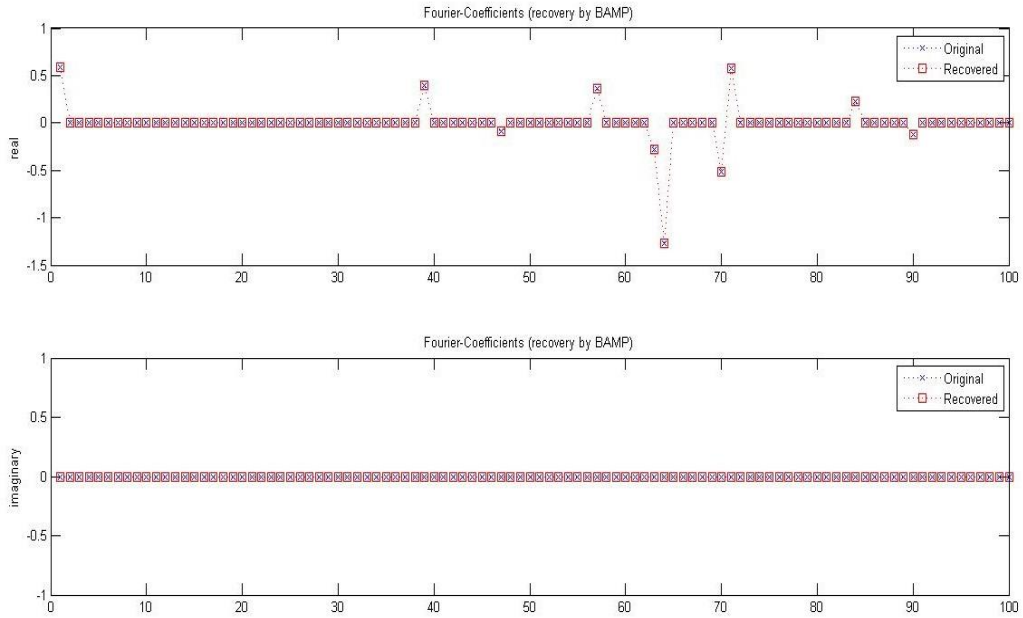


Figure 5.2 Comparison between Fourier Coefficients of a graph signal (symmetric weight matrix) recovered by the BAMP algorithm and the original coefficients. (Upper plot: real components, Lower plot: imaginary components).

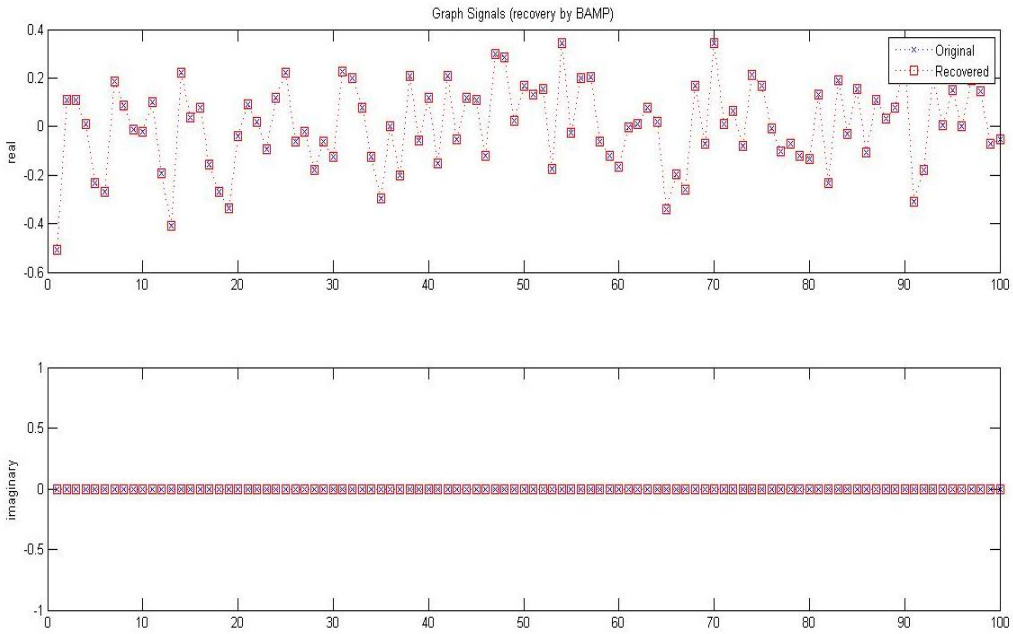


Figure 5.3 Comparison between a graph signal (symmetric weight matrix) recovered by BAMP and the original one (Upper plot: real components, Lower plot: imaginary components). The graph signal is constructed from the Fourier coefficients from Figure 5.2

A symmetric weight matrix was used, which means that after performing singular-value decomposition on this matrix, there will be only real eigen values. This can be seen in the results of the simulations shown in Figure 5.2. If an unsymmetrical weight matrix is used, we will get imaginary eigen values. This will be more complex to reconstruct the original signal using the given BAMP algorithm.

5.2 Quality of reconstruction

In the previous simulation, the ratio between M and N used was 1:2 or $\frac{M}{N} = 0.5$ and K was much smaller than M. The questions now are what happens if different dimensions were used and what parameter setting are needed to give us a good measure of the signal reconstruction.

In this thesis, the performance of reconstruction is defined as the logarithmic ratio between the energies of the original signal and the difference between the original signal and the estimated one

$$Quality\ of\ Reconstruciton|_{dB} = \frac{\sum_{j=1}^{N} {x^2}_j}{\sum_{j=1}^{N} \left({x^2}_j - {\hat{x}^2}_j\right)^2} \tag{5.5}$$

where x_j is the j component of the N-dimensional graph signal x while $\hat{x}_j$ is the j component of the reconstructed graph signal.

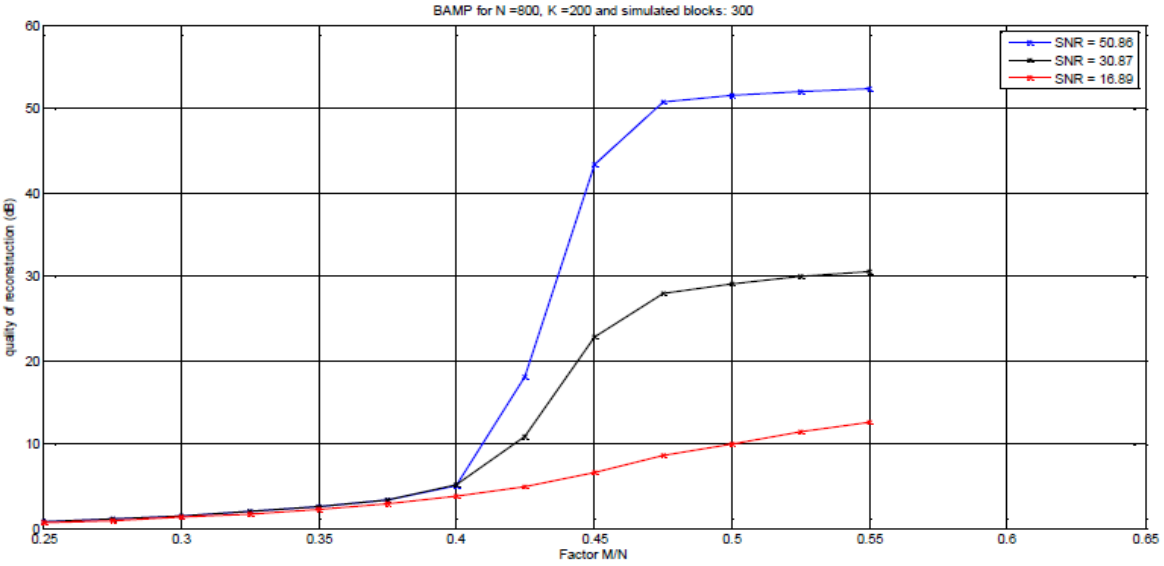
5.2.1 Varying Number of samples (M)

Two different sensing matrices were used for these simulations.

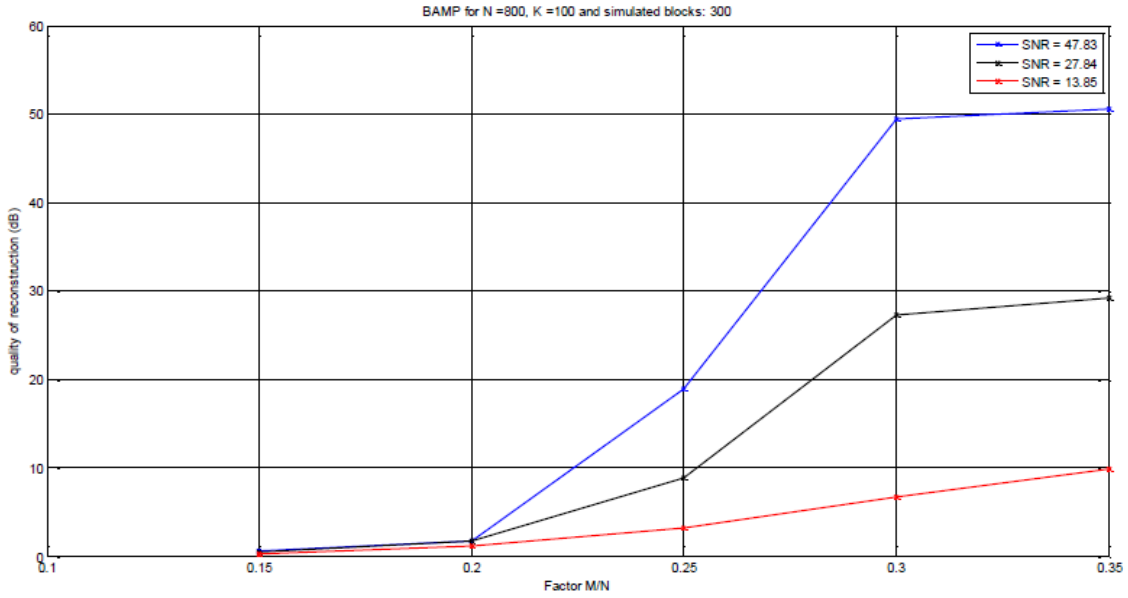
5.2.1.1 Singular-value Decomposition (SVD) Matrix

In the following simulation, we will check the effect of varying the number of samples (M) taken from the original signal on the quality of reconstruction. The simulation is repeated 300 times and the average of the quality measure in (5.5) from these simulations is used as a figure of merit. In Figure 5.4, the total number of signal

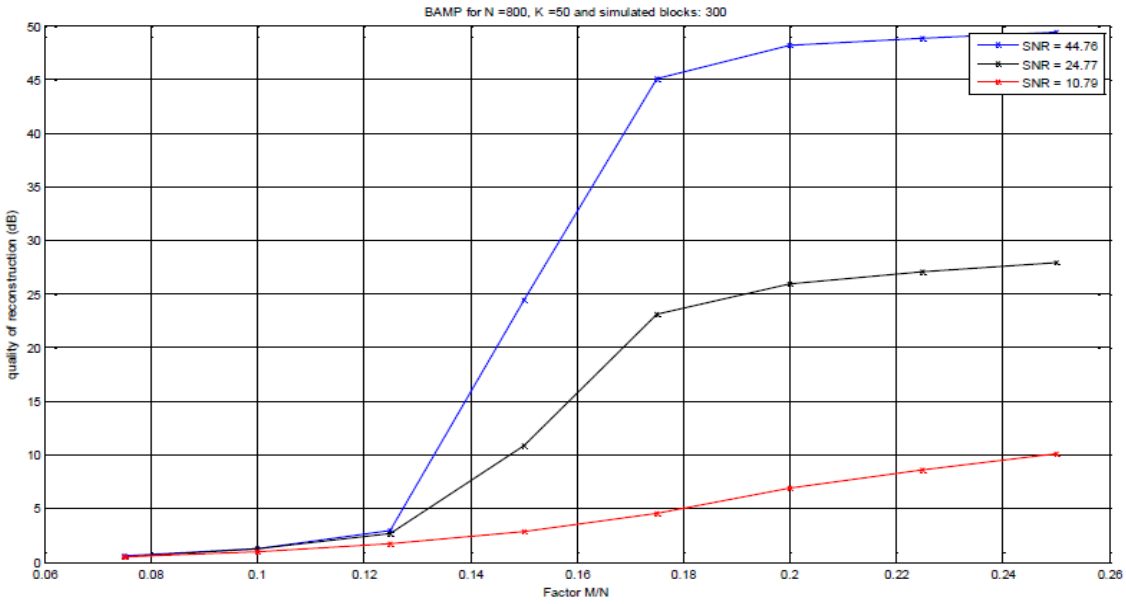
components *N* is fixed at 800 and for convenient representation of the results, it is stated as the ratio between M and N. The sensing matrix is produced using a singular-value decomposition of the weight matrix as mentioned earlier. Gaussian noise was added to all graph signals to investigate different signal to noise ratio (SNR) scenarios.



(a)



(b)



(c)

Figure 5.4 The effect varying $\frac{M}{N}$ versus the quality of reconstruction for different SNRs where N = 800 and in (a) K = 200, (b) K = 100 and in (c) K = 200.

As shown is Figure 5.4, as we increase the number of samples M, the quality of reconstruction gets better. After a certain point (transition point), it is observed that there is a significant improvement of the reconstruction quality. For example at (a) this transition point is when the $\frac{M}{N}$ equals 0.4, which means that M is almost twice the number of non-zero coefficients of the Fourier transform vector K and that is because $M \geq 2K$ is a necessary condition due to elementary null-space considerations of the compressed sensing problem. The same thing is seen in (b) and (c). Theorem 1 in Chapter 2, that M has to be greater than K does not apply here, as the locations of the K coefficients are unknown so the condition $M = K$ is not applied and the problem rather considered a compressed sensing problem for a band limited graph signal. For we have $M > 2K$ form basic null-space considerations as discussed in Chapter 2.

The quality of reconstruction will stop increasing and saturate after a certain point even if M is increased. Also if we compare (a), (b) and (c), the smaller K is the smaller M needs to be to achieve better quality of reconstruction. Finally, as the noise increases it is observed a strong deterioration of the quality of reconstruction, and with very high noise i.e. low SNR, there will not be a real transition point and the quality of reconstruction will experience a minimal improvement with increasing M.

This analysis can make us divide the plots into three regions, as shown in Figure 5.5. Region (1), which is before the transition point, where the samples are not enough to give us a very good quality of reconstruction. Region (2) is where the strongest increase in performance of the reconstruction is achieved when increasing M and then in Region (3) the performance saturates, as the SNR of the measurement noise becomes the limiting factor.

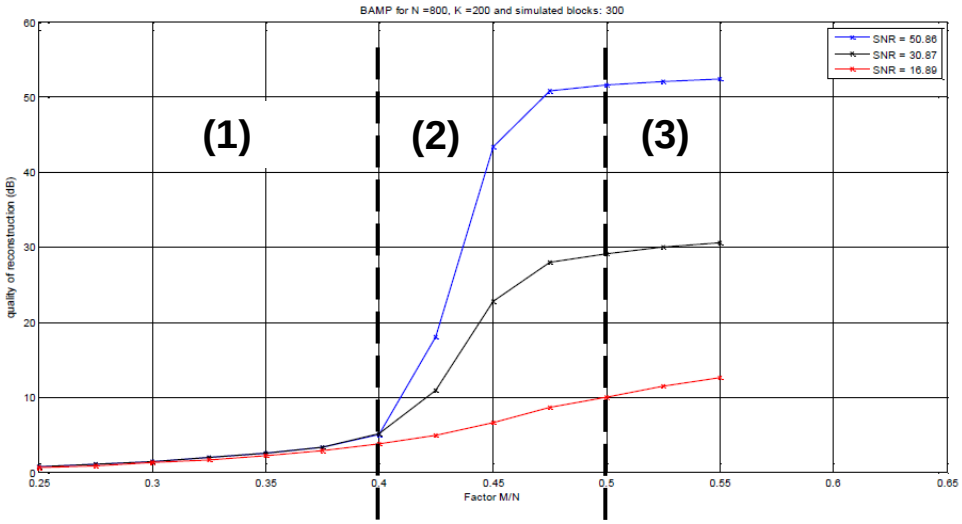


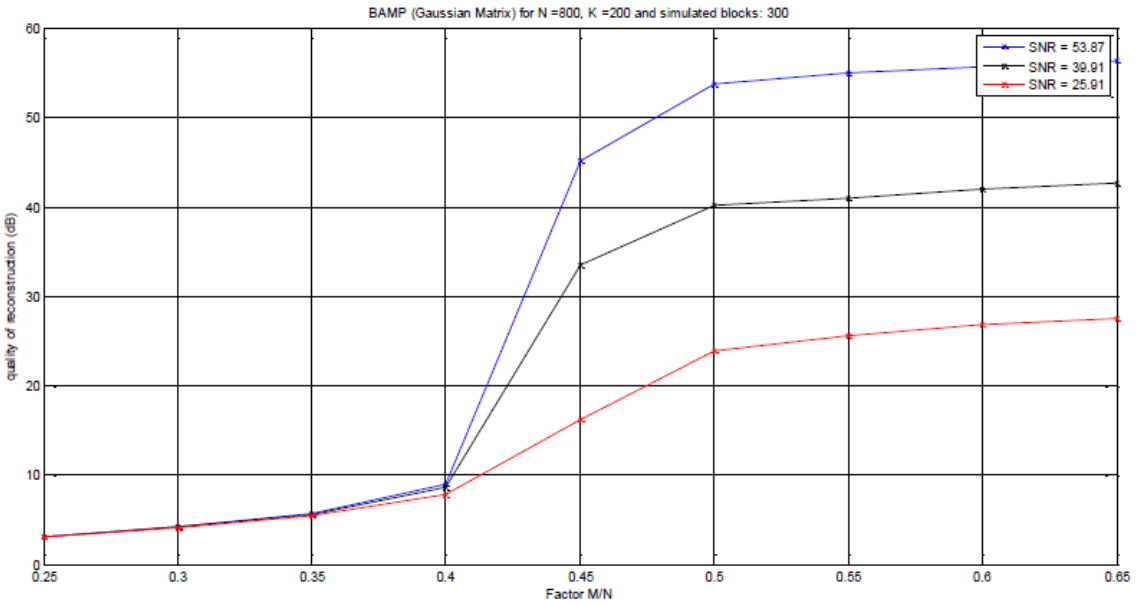
Figure 5.5 The effect varying $\frac{M}{N}$ versus the quality of reconstruction for different SNRs where N = 800 and the plot is divided to three regions for analysis.

The same simulations were repeated but using a signal with a smaller dimension N (See Appendix B). This new simulation gave different results than the ones shown in Figure 5.6 and that is: with smaller signal dimension N, a higher M: N ratio is required for better quality of reconstruction. If we compare Figure 5.4 (b) and Figure B.1 (b) where both of them have the same $K = 100$, we will find out that the transition point of the second one is at 0.50 while the first one at 0.2.

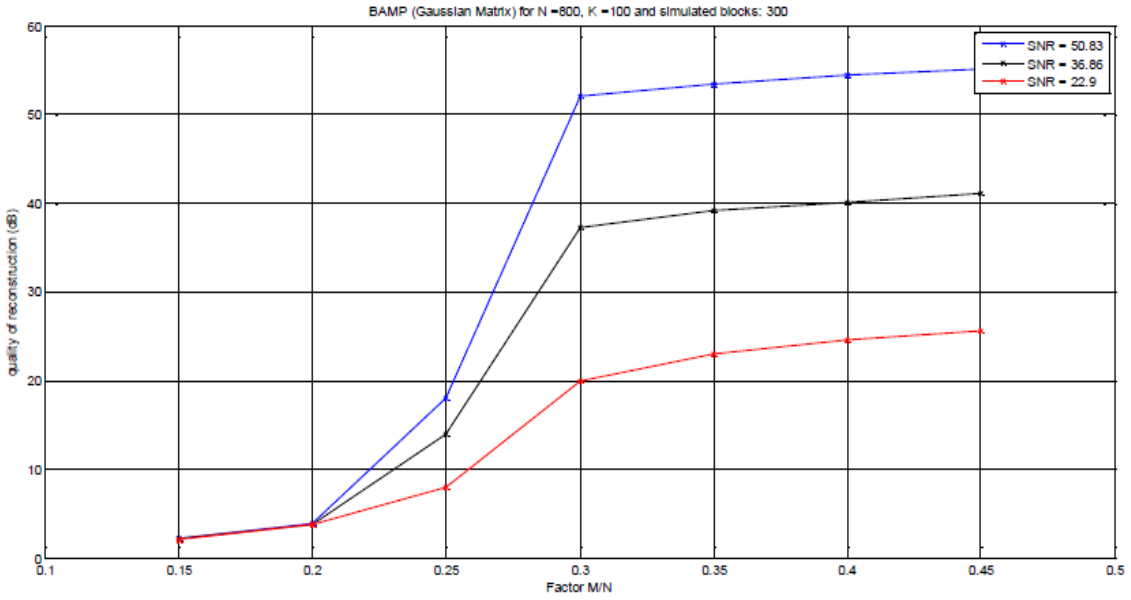
5.2.1.2 Gaussian Matrix

The sensing matrix plays an important role in the compressed sensing process, as mentioned in Chapter 3. This is why the previously done simulations were repeated, using a different sensing matrix. The components of the new sensing matrix are Gaussian random variables with zero mean. The Gaussian matrix would represent an ideal case from a

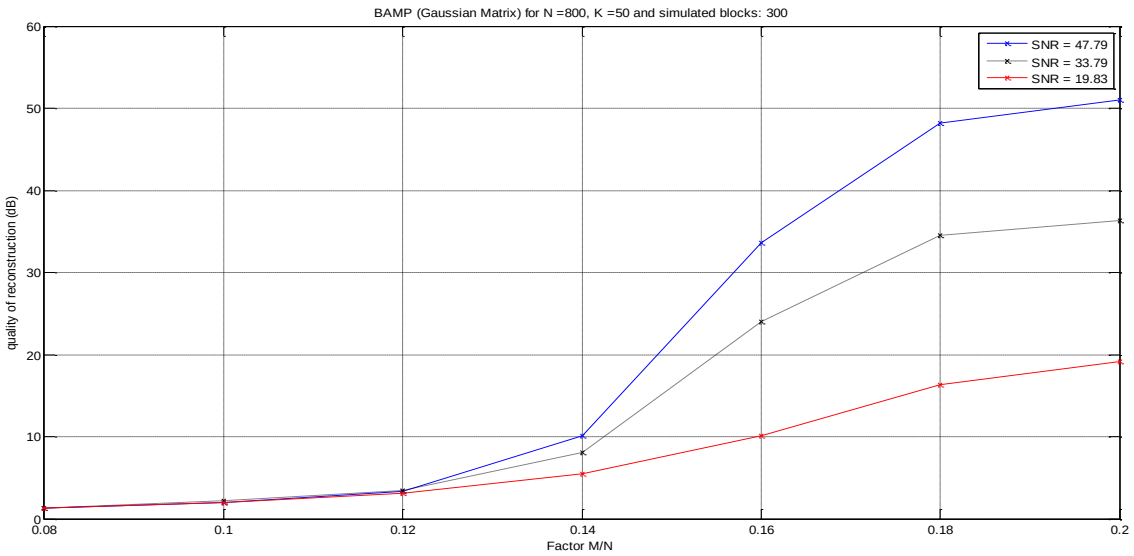
compressed-sensing perspective, so it would give us a good assessment to understand if the systematic construction of our graph sampling matrix from the eigen system of the weight matrix produces a loss relative to the ideal of a Gaussian random construction.



(a)



(b)



(c)

Figure 5.6 The effect varying $\frac{M}{N}$ versus the quality of reconstruction for different SNRs where N = 120 using the Gaussian matrix as the sensing matrix.

Comparing the results shown in Figure 5.6 to the results obtained previously in Figure 5.4 we will notice that they are very similar, which means that there was no loss in performance of signal recovery using the SVD matrix.

5.2.2 Varying the SNR

To get an overview of the performance of reconstruction over a wide range of SNR's, a simulation was made. A value for M was chosen from each region defined in Figure 5.5 and the same problem setup mentioned previously was used. As shown in Figure 5.7, the M chosen from region (1) didn't show a good performance even if the SNR was increased. The M from region (2) and (3) showed a significant increase in the performance after 12 dB SNR. M from region (3) has slightly better performance than the one from region (2) as the number of samples used is greater, making the estimation problem easier. Same simulations were carried out for different N's and these results are found in Appendix B.

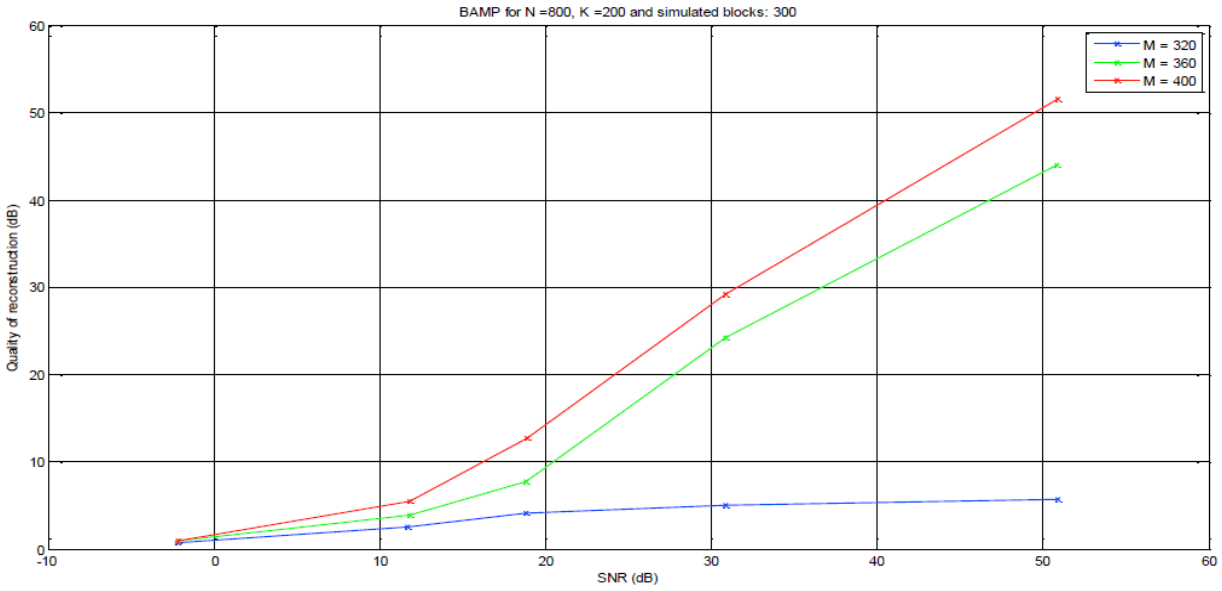


Figure 5.7 The effect varying SNR versus the quality of reconstruction for different M's.

Also a simulation was done to compare the quality of reconstruction using the SVD matrix and the Gaussian matrix as the sensing matrix. We can see from Figure 5.8 that both gave almost the same performance and no noticeable differences could be detected from this simulation.

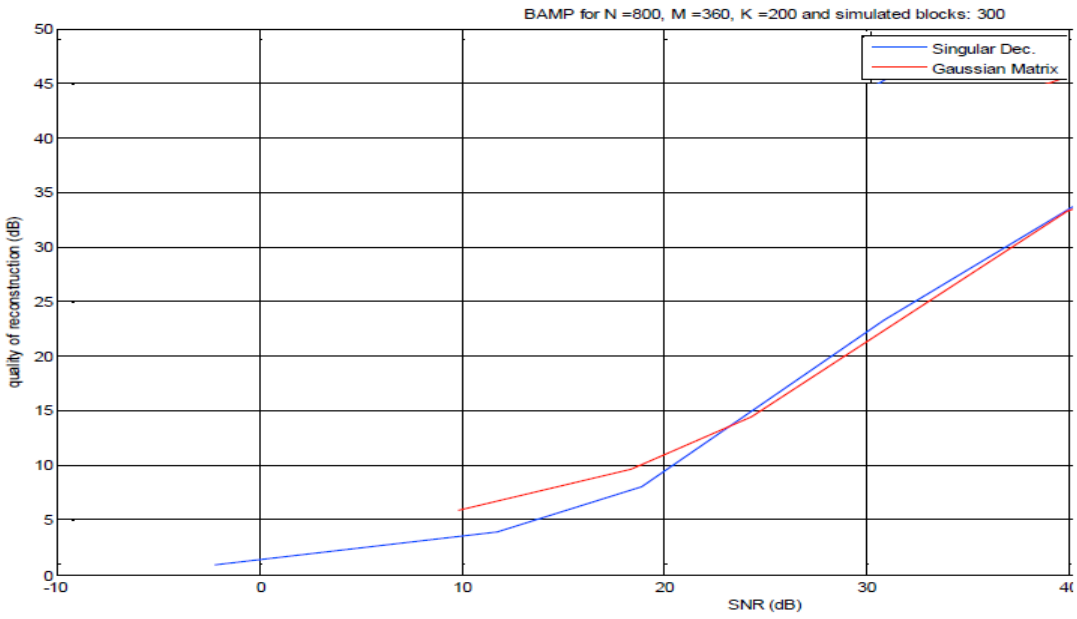


Figure 5.8 The effect varying SNR versus the quality of reconstruction for different M's and different sensing matrices.

5.3 Coherence of sensing Matrix

It was also important to get some analysis on the properties of the sensing matrices used in the previous simulations. As mentioned in Chapter 3, the RIP property is unfeasible to calculate in practice, so we checked the Coherence of the matrix. According to (3.10) the coherence of the matrix was calculated, also the lower bound (Walsh bound) was calculated.

5.3.1 SVD Matrix

The SVD sensing matrix is obtained as mentioned in Section 5.1. 300 simulations were done, and from these simulations the maximum, minimum and the average coherence was calculated as shown in Figure 5.9. Doing so, shows if the matrix has large variation in its coherence value or not, thus can be used conveniently.

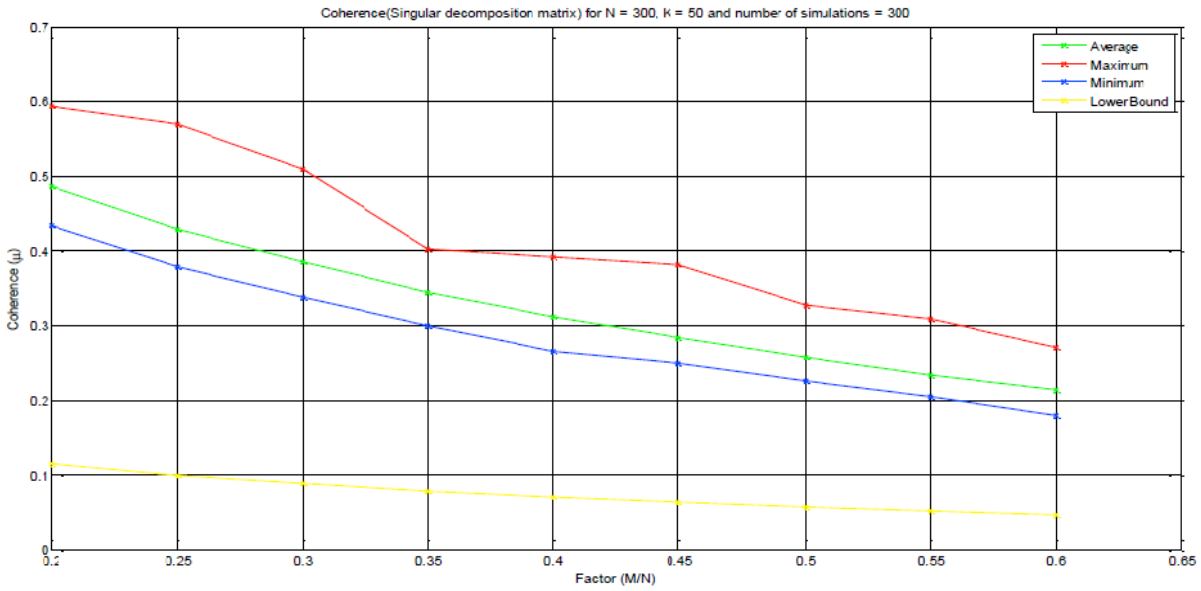


Figure 5.9 Coherence values of a SVD matrix for different dimensions.

5.3.2 Gaussian Matrix

The same procedure was repeated but using a Gaussian matrix. Based on (3.11) and (3.12) the lower the coherence the better the quality of reconstruction. In Figure 5.9 and 5.10, it is noticed that, as the number of samples taken (M) increases, the coherence value of the

matrix decreases, thus better quality of reconstruction and that matches what was concluded from (3.11) and (3.12).

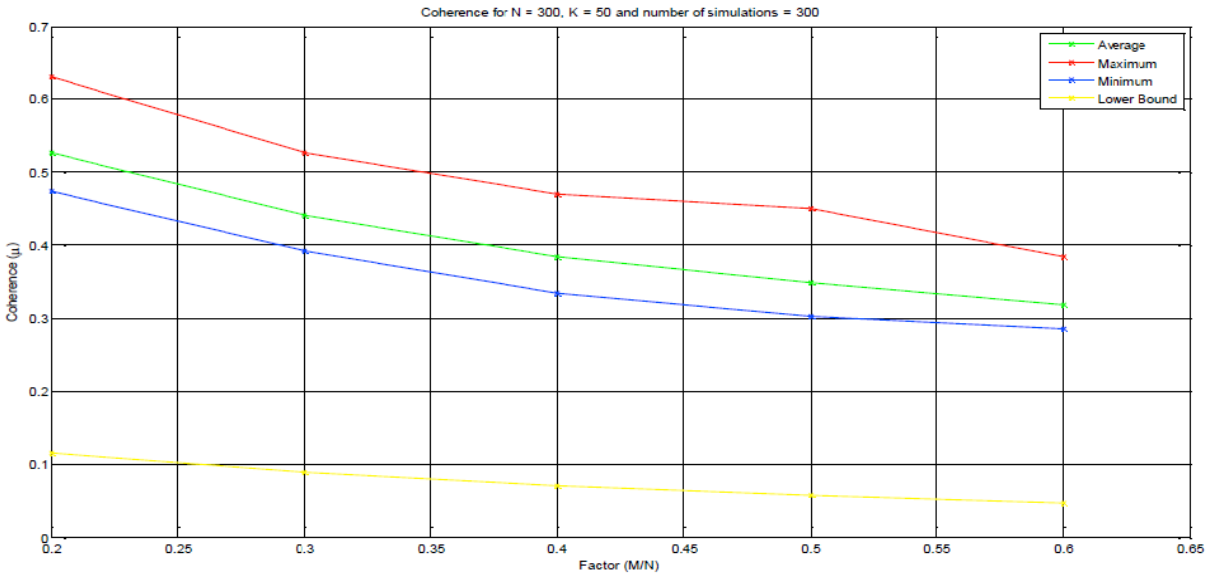


Figure 5.10 Coherence values of a SVD matrix for different dimensions.

5.3.3 Comparison between matrices

As shown in Figure 5.11, a comparison is made between the average coherence value of the SVD and Gaussian matrices. Although the performance of both matrices in signal reconstruction was almost the same, as shown previously, the coherence value of the Gaussian matrix is slightly higher than that of the SVD matrix. This is considered an interesting result that should be furtherly researched in future works.

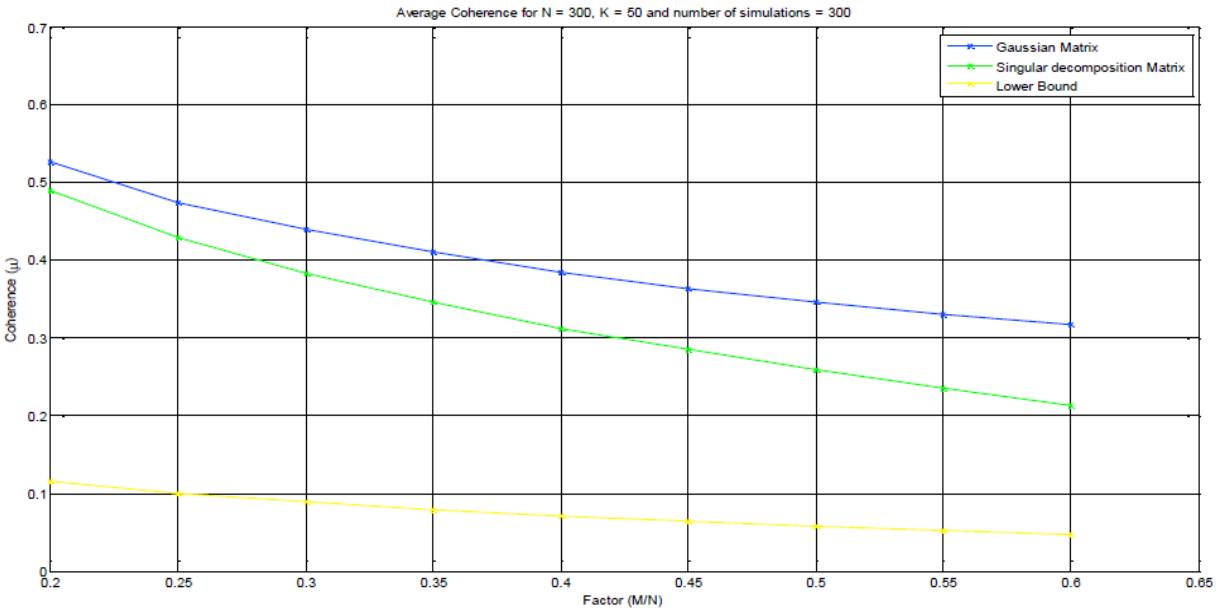


Figure 5.11 Comparison between the coherence values of a SVD and Gaussian matrix for different dimensions.

Chapter 6

Conclusion and Outlook

Graph signals gives us the ability to represent complex structured signals and carry out all the classical signal processing techniques like sampling, Fourier transform and others also for graph signals. In this thesis we tackled the problem of estimating a sparse graph signal with an unknown frequency support set, yet still we were able to reconstruct the original signal from a sampled noisy version. Not knowing the location of the non-zero Fourier coefficients and taking a number of samples more than $2K$ resulted in having a compressed sensing problem.

6.1 Conclusion

The BAMP algorithm was used to solve the compressed sensing problem and iteratively estimate the graph signal where the noise variance is unknown. The sensing matrix used was generated using a singular value decomposition of the weight matrix. After many simulations of different scenarios we found out that the more samples M used, the better the quality of reconstruction. The quality of reconstruction improved significantly starting from $M \geq 2K$ and this is due to elementary null-space considerations of the compressed sensing problem which can also be proved algebraically. Also this is considered a practical advantage, where the graph signal dimension N is relatively high, yet M can remain the same as used for a lower dimension as long as it satisfies this condition.

To check the potential loss in the reconstruction process produced due to using the SVD matrix, we used the Gaussian matrix which is considered the ideal case. The SVD

showed a similar performance to the Gaussian matrix. Also the Coherence property of these two matrices was calculated and turned out to be almost the same.

6.2 Future Work

The coherence of the Gaussian matrix was a bit larger than the coherence of the SVD matrix which would be interesting to analyze more in future works. It was mentioned in Chapter 3 that calculating the RIP of the sensing matrix is infeasible practically but there are certain random constructions of the matrix that can guarantee RIP with high probability, so it would be interesting to work more on this aspect. Also the case in which the sampling support set is unknown, would be an interesting topic to work on in future works, as the problem will be no more a compressed sensing problem.

Bibliography

[1] Siheng Chen, Rohan Varma, Aliaksei Sandryhaila, Jelena Kovacevic, *Discrete Signal Processing on Graphs:Sampling Theory*, IEEE Transactions on Signal Processing (Volume: 63, Issue: 24, Dec.15, 2015) Page(s): 6510 - 6523, August 2015

[2] Norbert Goertz, *Sampled Graph-Signals: Iterative Recovery with an Analytic Error Bound*, IEEE International Symposium on Information Theory, 2017

[3] Siheng Chen, Aliaksei Sandryhaila, Jose M. F. Moura, Jelena Kovacevic, *Signal Recovery on Graphs*, IEEE Transactions on Signal Processing (Volume: 63, Issue: 17, Sept.1, 2015) Page(s): 4609 – 4624, November 2011.

[4] Yi Gao, Jigen Peng, Shigang Yue and Yuan Zhao, *Research Article On the Null Space Property of l_q-Minimization for $0 < q \leq 1$ in Compressed Sensing*, Journal of Function Spaces Volume 2015 (2015), Article ID 579853, 10 pages http://dx.doi.org/10.1155/2015/579853 , March 2015

[5] M. A. Davenport, M. F. Duarte, Y. C. Eldar, and Gitta Kutyniok, *Introduction to Compressed Sensing*, http://statweb.stanford.edu/~markad/publications/ddek-chapter1-2011.pdf

[6] R. Gribonval andM. Nielsen, *Sparse representations in unions of bases*, IEEE Transactions on Information Theory, vol. 49, no. 12, pp. 3320–3325, 2003.

[7] MYUNG CHO, KUMAR VIJAY MISHRA and WEIYU XU, *New Algorithms for Verifying the Null Space Condition in Compressed Sensing*, https://arxiv.org/pdf/1604.02769v1.pdf , April 2016

[8] E. Candes and T. Tao., *Decoding by linear programming*, IEEE Transactions on Information Theory (Volume: 51, Issue: 12, Dec. 2005 Page(s): 4203 - 4215, November 2005.

[9] Emmanuel J. Cand`es and Yaniv Plan, *NEAR-IDEAL MODEL SELECTION BY $\ell 1$ MINIMIZATION*, The Annals statistics Volume 37, Number 5A (2009), 2145-2177, Aug 2009

[10] Tony Cai and Tiefeng Jiang, *Limiting Laws of Coherence of Random Matrices with Applications to Testing Covariance Structure and Construction of Compressed Sensing Matrices*, The Annals statistics Volume 39, Number 3 (2011), 1496-1525 , Feb 2011

[11] S. Sarvotham, D. Baron, and R. G. Baraniuk*, "Compressed sensing reconstruction via belief propagation,"* Rice University, Houston, TX, Tech. Rep. TREE0601, July 2006.

[12] Yanting Ma, Dror Baron, and Deanna Needell, *Two-Part Reconstruction with Noisy-Sudocodes*, IEEE Transactions on Signal Processing (Volume: 62, Issue: 23, Dec.1, 2014), Sep 2014

[13] David L. Donoho, Arian Maleki, and Andrea Montanari, *Message Passing Algorithms for Compressed*
Sensing: II. Analysis and Validation, Information Theory (ITW 2010, Cairo), 2010 IEEE Information Theory Workshop, July 2010

[14] Norbert Goertz*, A Pratical Guide to Bayesian-optimal Approximate Message Passing in Compressed Sensing*, Report TU Wien, 2016

Appendix A

BAMP with General Full-Rank Measurement Matrices [4.4]

Given the measurement equation:

$$\mathbf{y} = \mathbf{\Phi}\mathrm{s} + \mathbf{w} \tag{A.1}$$

with the vector y of dimension $m \times 1$ representing the observations taken and s $(n \times 1)$ the signal vector $= \{\mathrm{s}_j, j = 1, \dots, n\}$. The sensing matrix is:

$$\mathbf{\Phi} = \{\mathbf{\Phi}_1, \mathbf{\Phi}_2, \dots., \mathbf{\Phi}_n\} \tag{A.2}$$

of dimension $m \times n$ with $m < n$. $\mathbf{\Phi}$ is assumed to have a rank m but the columns are *not normalized*. $\mathbf{w}$ is the measurement noise and its components are independent and identically distributed Gaussian with variance $\sigma^2 > 0$.

The ℓ_2-norms of the columns of the measurement matrix are defined as:

$$\varphi_j = \left\|\mathbf{\Phi}_j\right\|_2 \qquad j = 1,2, \dots., n \tag{A.3}$$

So it makes (A.1) to be:

$$\mathbf{y} = \underbrace{\left\{\frac{\mathbf{\Phi}_1}{\varphi_1}, \frac{\mathbf{\Phi}_2}{\varphi_2}, \dots, \frac{\mathbf{\Phi}_n}{\varphi_n}\right\}}_{=\mathbf{A}} \underbrace{\begin{Bmatrix} \varphi_1 s_1 \\ \varphi_2 s_2 \\ \vdots \\ . \\ \varphi_n s_n \end{Bmatrix}}_{=\mathbf{X}} + \mathbf{w} \tag{A.4}$$

where $\mathbf{A}$ is the new measurement matrix with columns $\mathbf{A}_j = \frac{\mathbf{\Phi}_j}{\varphi_j}$ that are normalized according to $\left\|\mathbf{A}_j\right\|_2 = 1$. Now the measurement matrix fulfils the condition to be used for the BAMP algorithm, but still the signal prior has to be rescaled.

With the given signal prior $p_{S_j}(s_j)$, the pdf of the random variable $X_j = \varphi_j S_j$ is:

$$p_{X_j}(x_j) = \frac{1}{|\varphi_j|} p_{S_j}\left(\frac{x_j}{\varphi_j}\right) \tag{A.5}$$

For the Gaussian prior

$$p_{S_j}(s_j) = \frac{1}{\sqrt{2\pi}\sigma_{S_j}} e^{-\frac{1}{2\sigma^2 S_j}{s_j}^2} \tag{A.6}$$

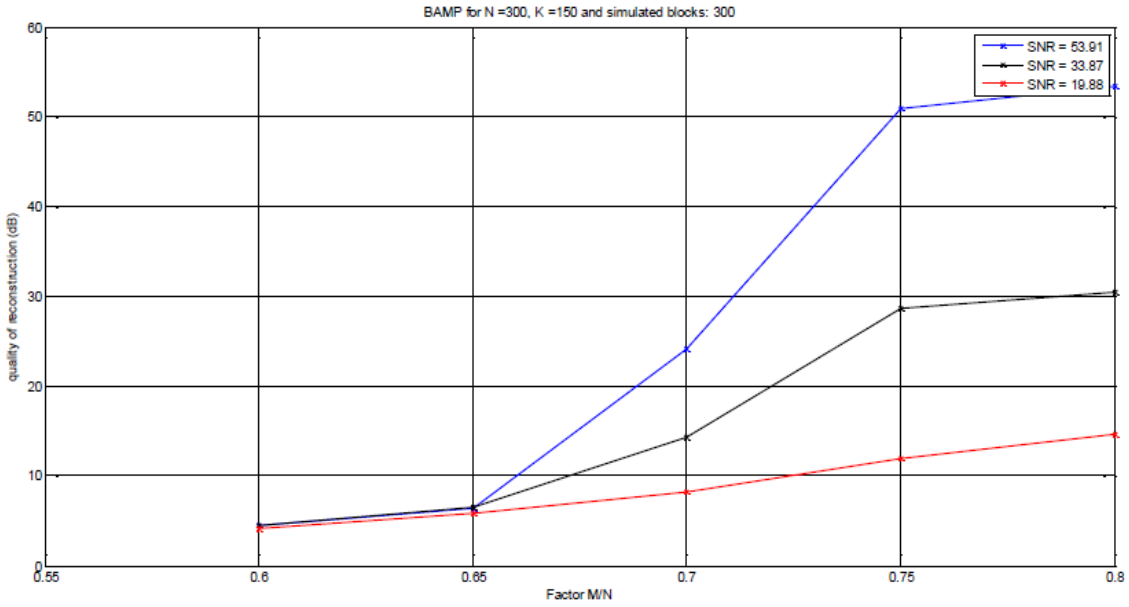
we obtain

$$p_{X_j}(X_j) = \frac{1}{\varphi_j}\frac{1}{\sqrt{2\pi}\sigma_{S_j}} e^{-\frac{1}{2\sigma^2 S_j\ {\varphi^2}_j}\ {s^2}_j} \tag{A.7}$$

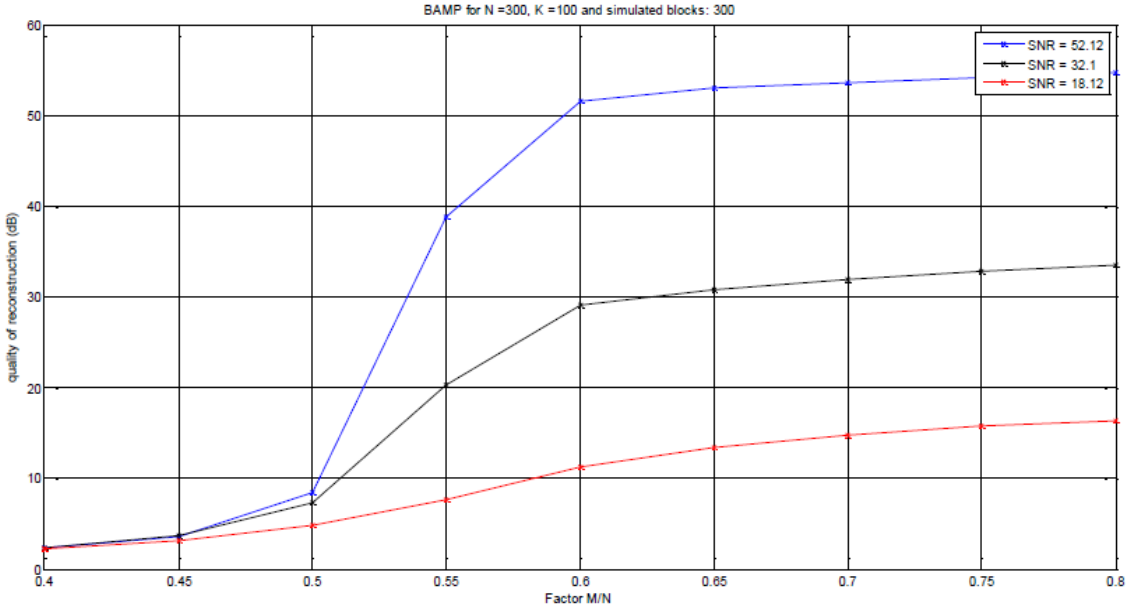
with scaled variance ${\sigma^2}_{X_j} = {\varphi^2}_j\ {\sigma^2}_{S_j}$.

Appendix B

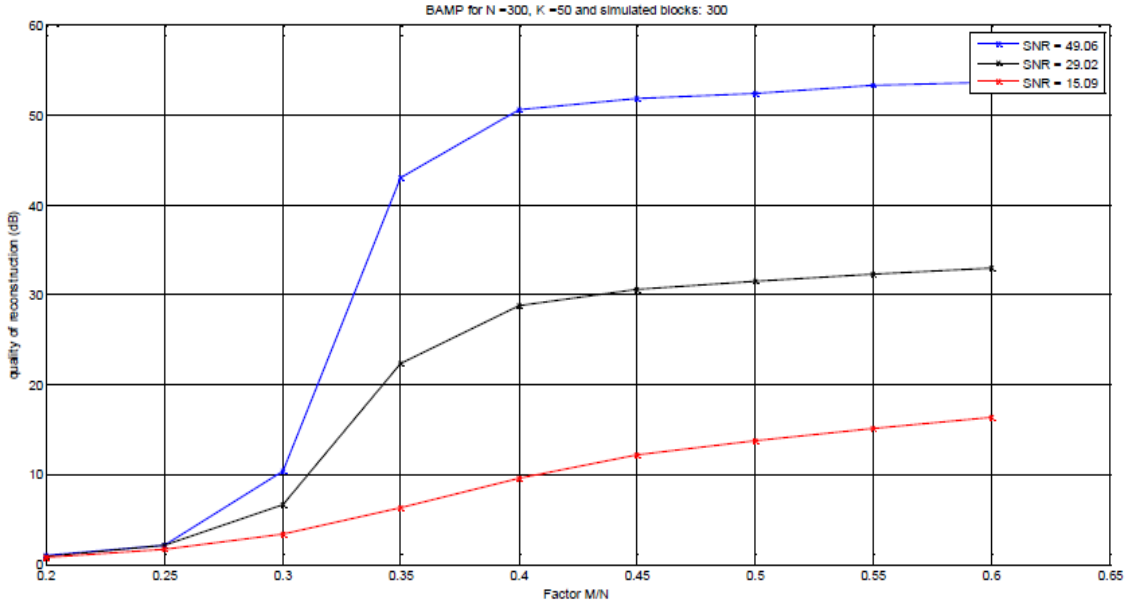
More Simulation results



(a)

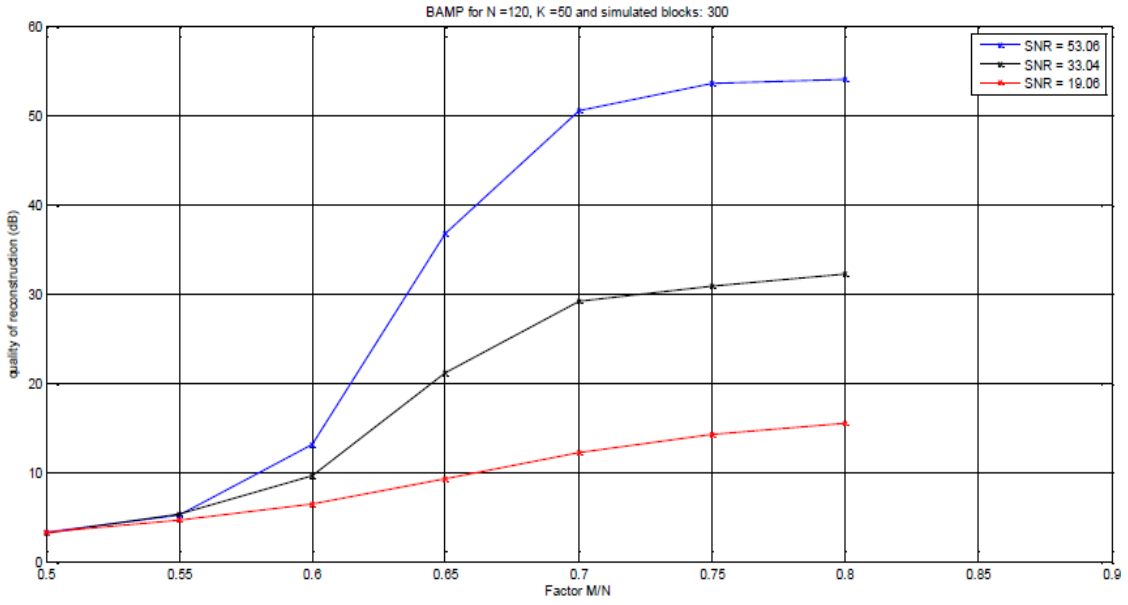


(b)

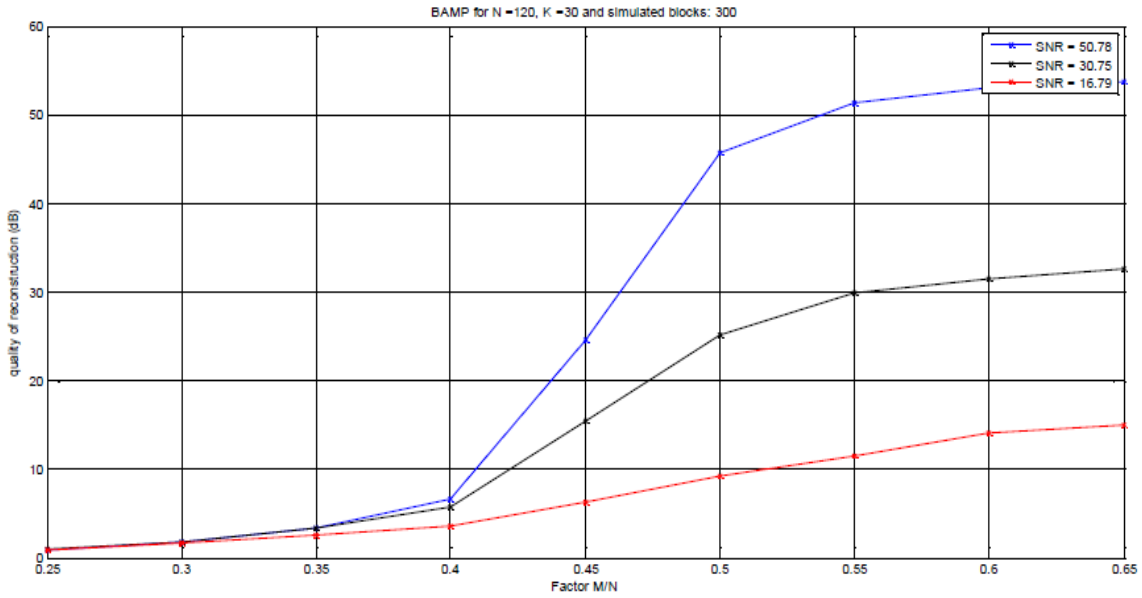


(c)

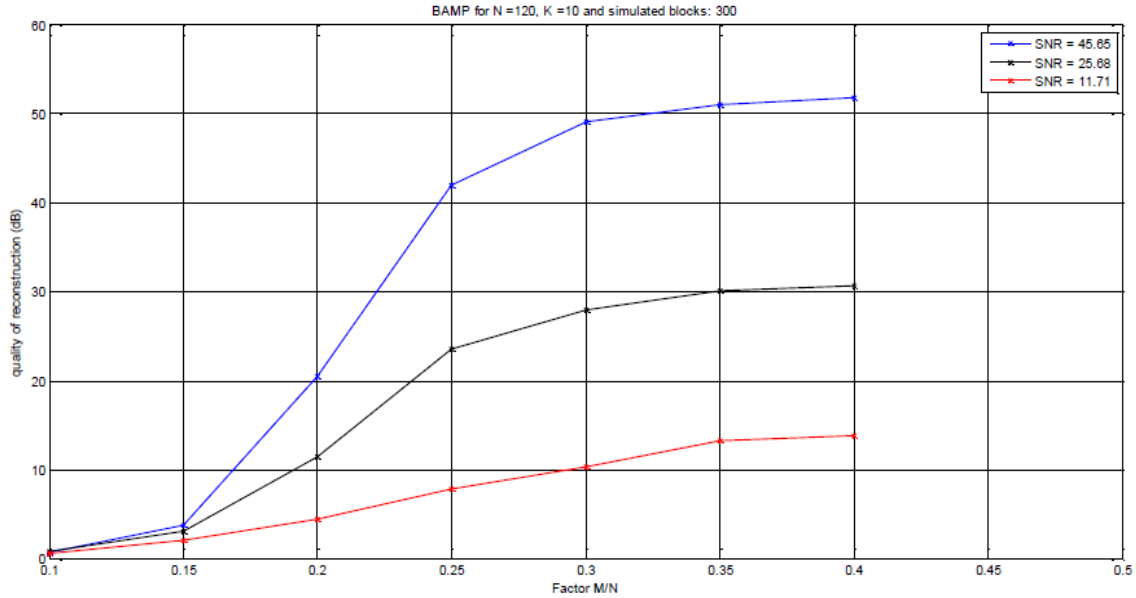
Figure B.1 The effect varying $\frac{M}{N}$ versus the quality of reconstruction for different SNRs where N = 300.



(a)

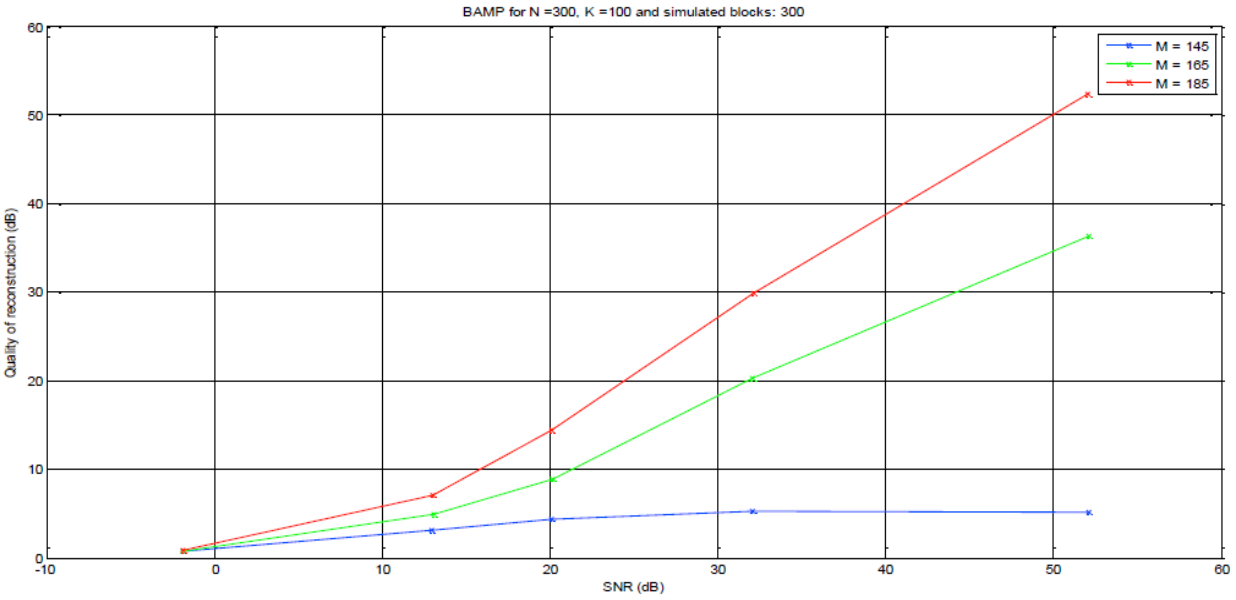


(b)

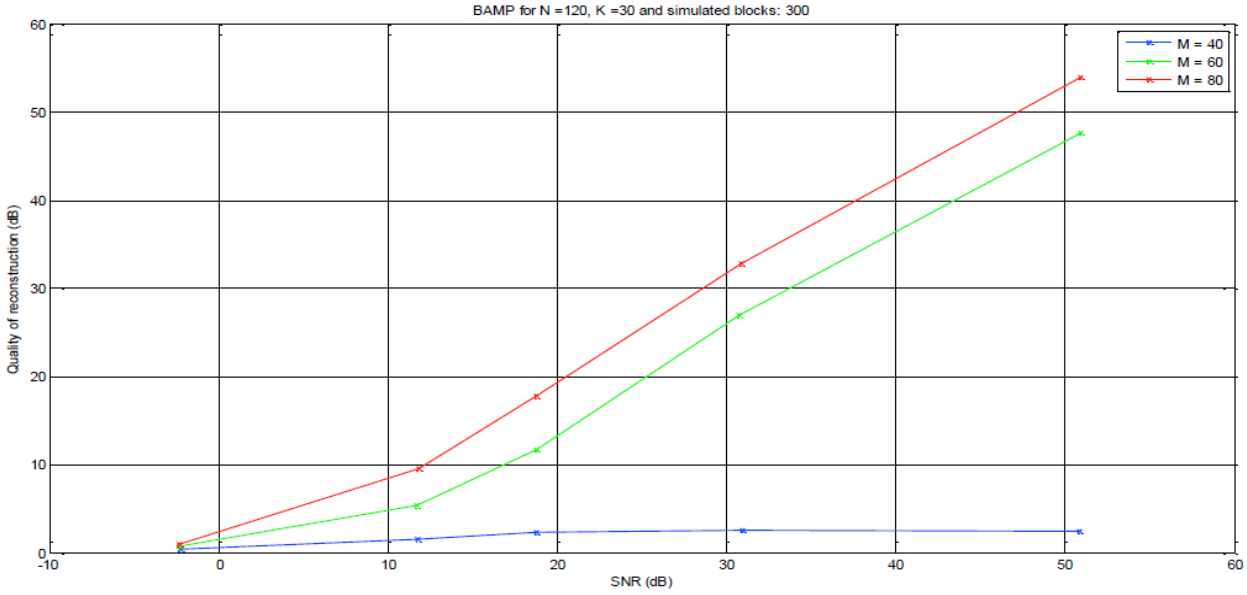


(c)

Figure B.2 The effect varying $\frac{M}{N}$ versus the quality of reconstruction for different SNRs where N = 120.



(a)



(b)

Figure B.3 The effect varying SNR versus the quality of reconstruction for different M's.