LDPC Codes for Compressive Sensing

Akshit Kumar[1] Sachin Agrawal[2]

[1]Department of Electrical Engineering
Indian Institute of Technology, Madras

[2]Department of Electrical Engineering
Indian Institute of Technology, Madras

Outline

Agenda

- Review Compressed Sensing along with the linear programming relaxation, **CS-LPD**

Agenda

- Review Compressed Sensing along with the linear programming relaxation, **CS-LPD**
- Review Channel Coding along with the linear programming relaxation, **CC-LPD**

Agenda

- Review Compressed Sensing along with the linear programming relaxation, **CS-LPD**
- Review Channel Coding along with the linear programming relaxation, **CC-LPD**
- Interpretation of the bridge connection between **CC-LPD** and **CS-LPD**

Agenda

- Review Compressed Sensing along with the linear programming relaxation, **CS-LPD**
- Review Channel Coding along with the linear programming relaxation, **CC-LPD**
- Interpretation of the bridge connection between **CC-LPD** and **CS-LPD**
- Illustrate the results with example of parity check matrix of (7,4) Hamming Code

Outline

Motivation

- Consider the classical linear algebra problem of solving $y = Ax$ for some *measurement matrix* $A \in \mathbb{R}^{mxn}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. So given y, can we recover x?

Motivation

- Consider the classical linear algebra problem of solving $y = Ax$ for some *measurement matrix* $A \in \mathbb{R}^{mxn}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. So given y, can we recover x?
- The question is when can we solve this linear system of equations?

Motivation

- Consider the classical linear algebra problem of solving $y = Ax$ for some *measurement matrix* $A \in \mathbb{R}^{mxn}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. So given y, can we recover x?
- The question is when can we solve this linear system of equations?
- In case $m \geq n$ and A has full rank, then we have an over-determined system and can be solved uniquely, say by using gaussian elimination. Though in practice this is not an interesting case. Rarely do we have more measurements than we need.

Motivation

- Consider the classical linear algebra problem of solving $y = Ax$ for some *measurement matrix* $A \in \mathbb{R}^{mxn}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. So given y, can we recover x?
- The question is when can we solve this linear system of equations?
- In case $m \geq n$ and A has full rank, then we have an over-determined system and can be solved uniquely, say by using gaussian elimination. Though in practice this is not an interesting case. Rarely do we have more measurements than we need.
- In case $m < n$ and A is full rank, the system is under-determined system and can have no solutions or infinite solutions.

Motivation

- Consider the classical linear algebra problem of solving $y = Ax$ for some *measurement matrix* $A \in \mathbb{R}^{mxn}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. So given y, can we recover x?
- The question is when can we solve this linear system of equations?
- In case $m \geq n$ and A has full rank, then we have an over-determined system and can be solved uniquely, say by using gaussian elimination. Though in practice this is not an interesting case. Rarely do we have more measurements than we need.
- In case $m < n$ and A is full rank, the system is under-determined system and can have no solutions or infinite solutions.
- The under-determined system seems like an impossible problem to solve. Can we do better?

Outline

Sparse Recovery

- For most practical cases, our signal x is "spiky" i.e. it has non-zero values only in few of it's coordinates.

Sparse Recovery

- For most practical cases, our signal x is “spiky” i.e. it has non-zero values only in few of it’s coordinates.

Example

Consider the sinusoidal function, say $x(t) = sin(\omega t)$, in time domain the signal takes continuous values but in the frequency domain, it can be represented as a “spiky” signal by taking the Fourier transform

Definition

A signal x is said to be s-sparse if at most s coordinates of x are non-zero.

Does Sparsity Help?

The central idea to the field of compressive sensing theory is the following:

Theorem

If every $2s$ distinct columns in A are linearly independent (assuming $m \geq 2s$), then any s-sparse signal can be reconstructed uniquely from Ax.

Proof.

Suppose, there exist two s-sparse signals $x, x^{'} \in \mathbb{R}^n$ such that $Ax = Ax^{'}$,then $A(x - x^{'}) = 0$. Now $x - x^{'}$ is $2s$-sparse,so there is a linear dependence between $2s$ columns of A, hence a contradiction. $\square$

Constructing x from $y = Ax$

- The previous proof shows how to construct x : x is the *unique sparsest* solution to the linear system of equations $y = Ax$, in other words we can formulate it as an optimization problem

Constructing x from $y = Ax$

- The previous proof shows how to construct x : x is the *unique sparsest* solution to the linear system of equations $y = Ax$, in other words we can formulate it as an optimization problem

$$\begin{aligned} &\min_x \|x\|_0 \\ &\text{s.t. } A.x = y \end{aligned} \tag{1}$$

- ℓ_0 minimization is an NP-hard problem and is computationally intractable.

Basis Pursuit

- We can relax the ℓ_0-minimization problem by ℓ_1-minimization problem.
- This is a convex optimization problem and can be solved in polynomial time by linear programming methods.
- Intuition for relaxing it to an ℓ_1 minimization problem is given below :

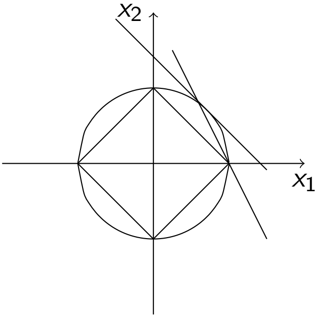


Figure: ℓ_1 minimization vs ℓ_2 minimization

Formalizing the Compressive Sensing Problem

- Let us consider a measurement matrix H_{CS} and a system of equations given by $y = H_{CS}x$. The compressive sensing problem boils down to the following problem : Given y, recover x assuming x is s-sparse.
- The optimal solution to the problem can be achieved by solving the following minimization problem, to be referred as **CS-OPT**.

$$\begin{aligned} &\min_x \|x\|_0 \\ &\text{s.t. } H_{CS}.x = y \end{aligned} \tag{CS-OPT}$$

- Since **CS-OPT** is an NP-hard problem,we relax it and refer to it as **CS-LPD**

$$\begin{aligned} &\min_x \|x\|_1 \\ &\text{s.t. } H_{CS}.x = y \end{aligned} \tag{CS-LPD}$$

So when is **CS-OPT** and **CS-LPD** equivalent?

- We relaxed **CS-OPT** to **CS-LPD**, but would the two optimization problems yield the same x?

So when is **CS-OPT** and **CS-LPD** equivalent?

- We relaxed **CS-OPT** to **CS-LPD**, but would the two optimization problems yield the same x?
- The answer is YES, under some conditions.

So when is **CS-OPT** and **CS-LPD** equivalent?

- We relaxed **CS-OPT** to **CS-LPD**, but would the two optimization problems yield the same x?
- The answer is YES, under some conditions.
- To characterize the condition, we need the *nullspace characterization* of "good" measurement matrices.

So when is **CS-OPT** and **CS-LPD** equivalent?

- We relaxed **CS-OPT** to **CS-LPD**, but would the two optimization problems yield the same x?
- The answer is YES, under some conditions.
- To characterize the condition, we need the *nullspace characterization* of "good" measurement matrices.
- A measurement matrix H_{CS} is said to be s-good if ℓ_1-minimization recovers correctly all s-sparse vectors x

So when is **CS-OPT** and **CS-LPD** equivalent?

- We relaxed **CS-OPT** to **CS-LPD**, but would the two optimization problems yield the same x?
- The answer is YES, under some conditions.
- To characterize the condition, we need the *nullspace characterization* of "good" measurement matrices.
- A measurement matrix H_{CS} is said to be s-good if ℓ_1-minimization recovers correctly all s-sparse vectors x
- A necessary and sufficient condition for H_{CS} to be s-good is the following *nullspace property* :

$$\|v_I\|_1 < \frac{1}{2}\|v\|_1$$

for all $v \in \mathbb{R}^n : Av = 0, v \neq 0, I \subset \{1, ..., n\}, |I| \leq s$

So when is **CS-OPT** and **CS-LPD** equivalent?

Theorem

Let H_{CS} be a measurement matrix. Also, assume that $y = H_{CS}x$ and that x has at most k non zero elements i.e. $\|x\|_0 \leq k$. Then the estimate $\hat{e}$ given by **CS-LPD** *will be equal to the estimate $\hat{e}$ given by* **CS-OPT** *if H_{CS} satisfies the nullspace property.*

So when is **CS-OPT** and **CS-LPD** equivalent?

Theorem

Let H_{CS} be a measurement matrix. Also, assume that $y = H_{CS}x$ and that x has at most k non zero elements i.e. $\|x\|_0 \leq k$. Then the estimate $\hat{e}$ given by **CS-LPD** *will be equal to the estimate $\hat{e}$ given by* **CS-OPT** *if H_{CS} satisfies the nullspace property.*

- In conclusion, the estimate of **CS-OPT** and **CS-LPD** are equal as long as the H_{CS} is s-good.

Outline

Practical Application of Compressive Sensing

- Areas of Application of Compressive Sensing include

Practical Application of Compressive Sensing

- Areas of Application of Compressive Sensing include
  - Astronomy
  - Biology
  - Communications
  - Image and Video Processing
- Exciting application of Compressive Sensing include:

Practical Application of Compressive Sensing

- Areas of Application of Compressive Sensing include
  - Astronomy
  - Biology
  - Communications
  - Image and Video Processing
- Exciting application of Compressive Sensing include:
  - MRI
  - Single Pixel Camera

Outline

Block codes

H_{CC}: Parity check matrix of dimension r x m
A vector $c \in \{0,1\}^n$ *lies in the codespace if and only if*

$$H_{CC}.c = 0$$

The codespace($\mathcal{C}$)=$\{x| < h_i, x >= 0\ \forall i \in \{1, \ldots, m\}\}$

The MLD formulation

- Log likelihood ratio of ith bit is:

$$\gamma_i = ln\left(\frac{Pr[r_i|y_i = 0]}{Pr[r_i|y_i = 1]}\right)$$

The MLD formulation

- Log likelihood ratio of ith bit is:

$$\gamma_i = ln\left(\frac{Pr[r_i|y_i = 0]}{Pr[r_i|y_i = 1]}\right)$$

- Seen as a cost of assigning a bit value of 1.
- Let f_i denote the LP variables over which we minimise cost. There is one variable corresponding to each bit.

Equivalent MLD formulations

- $$min_f \sum_{i=1}^{n} \gamma_i f_i \qquad \text{(CC-MLD)}$$
  $$\text{s.t. } f \in \mathcal{C}$$

Equivalent MLD formulations

- $$\min_f \sum_{i=1}^{n} \gamma_i f_i \qquad \text{(CC-MLD)}$$
  $$\text{s.t. } f \in \mathcal{C}$$

- $$\min_f \sum_{i=1}^{n} \gamma_i f_i \qquad \text{(CC-MLD)}$$
  $$\text{s.t. } f \in conv(\mathcal{C})$$

- Since the optimal values of a convex optimisation problem occurs at the extreme points, we can say that the 2 problems are equivalent.

Equivalent MLD formulations

- $$\min_f \sum_{i=1}^{n} \gamma_i f_i \quad \text{s.t. } f \in \mathcal{C} \tag{CC-MLD}$$
- $$\min_f \sum_{i=1}^{n} \gamma_i f_i \quad \text{s.t. } f \in conv(\mathcal{C}) \tag{CC-MLD}$$
- Since the optimal values of a convex optimisation problem occurs at the extreme points, we can say that the 2 problems are equivalent.
- The convex hull has exponential description complexity.

Outline

Relaxing CC-MLD

- To relax the problem further we need the result:

$$conv(A \cap B) \subseteq conv(A) \cap conv(B)$$

Relaxing CC-MLD

- To relax the problem further we need the result:

$$conv(A \cap B) \subseteq conv(A) \cap conv(B)$$

- $S_j = \{x \mid H_{CC,j}.x = 0 (mod\ 2)\}$
- $$\begin{aligned} conv(C_{CC}) &= conv(S_1 \cap S_2 \cap \ldots S_m) \\ conv(C_{CC}) &\subseteq \ conv(S_1) \cap conv(S_2) \cap \ldots conv(S_m) \end{aligned} \tag{2}$$

CC-LPD

- $$\mathcal{P}(H_{CC}) = \cap_{i=1}^{m} conv(S_i)$$

  .

CC-LPD

- $$\mathcal{P}(H_{CC}) = \cap_{i=1}^{m} conv(S_i)$$

  .

- $$min_f \sum_{i=1}^{n} \gamma_i f_i \qquad \text{(CC-LPD)}$$
  $$\text{s.t. } f \in \mathcal{P}(H_{CC})$$

CC-LPD

- $$\mathcal{P}(H_{CC}) = \cap_{i=1}^{m} conv(S_i)$$
  .
- $$min_f \sum_{i=1}^{n} \gamma_i f_i \quad \text{s.t. } f \in \mathcal{P}(H_{CC}) \tag{CC-LPD}$$
- A relaxation to the original problem.

Illustration of H_{CC}

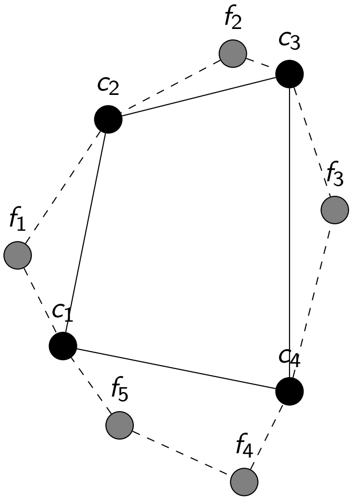


Figure: The darkened lines represent the convex hull of the codespace, and the dotted lines represent the polytope $\mathcal{P}(H_{CC})$. The gray nodes represent fractional

Outline

Fractional distance

- $$d_{frac} = min \sum_{i=1}^{N} |y_i - f_i| \qquad (3)$$
  $$\text{s.t } y \in \mathcal{C}, f \in \mathcal{V}(\mathcal{P}), y \neq f$$

Fractional distance

- $$d_{frac} = min \sum_{i=1}^{N} |y_i - f_i| \\ \text{s.t } y \in \mathcal{C}, f \in \mathcal{V}(\mathcal{P}), y \neq f \tag{3}$$

- It can then be shown that CC-LPD can correct upto $[\frac{d_{frac}-1}{2}]$ errors.

Outline

The fundamental cone

- *Definition:* The fundamental cone $\mathcal{K}(H_{CC})$ is defined as the set of vectors $\nu = (\nu_1, \nu_2, \ldots, \nu_n) \in \mathbb{R}^n$such that $\forall$ $1 \leq i \leq n$ and $1 \leq j \leq m$, we have the following:

$$\nu_i \geq 0$$

$$\sum_{i' \neq i} h_{ji'} \nu_{i'} \geq h_{ji} \nu_i$$

The fundamental cone

- *Definition:* The fundamental cone $\mathcal{K}(H_{CC})$ is defined as the set of vectors $\nu = (\nu_1, \nu_2, \ldots, \nu_n) \in \mathbb{R}^n$such that $\forall$ $1 \leq i \leq n$ and $1 \leq j \leq m$, we have the following:

$$\nu_i \geq 0$$

$$\sum_{i' \neq i} h_{ji'} \nu_{i'} \geq h_{ji} \nu_i$$

- $$H = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

- $\nu \in \mathcal{K}$ if: $\nu_1 = \nu_2 = \nu_3,\ \nu_5 = \nu_6 = \nu_7$
  $2\nu_1 \geq \nu_4,\ 2\nu_5 \geq \nu_4$

Condition for correct decoding

- *Lemma:* Let $\mathcal{S}$ denote the set of indices flipped over during transmission over a BSC channel. If the parity check matrix H_{CC} is such that for all vectors(ω) in the fundamental cone of the parity matrix, the following holds:

$$||\omega_{\mathcal{S}}||_1 < ||\omega_{\overline{\mathcal{S}}}||_1 \tag{4}$$

. If the above condition holds then CC-LPD will decode to the same point as CC-MLD.

Condition for correct decoding

- *Lemma:* Let $\mathcal{S}$ denote the set of indices flipped over during transmission over a BSC channel. If the parity check matrix H_{CC} is such that for all vectors(ω) in the fundamental cone of the parity matrix, the following holds:

$$||\omega_{\mathcal{S}}||_1 < ||\omega_{\overline{\mathcal{S}}}||_1 \tag{4}$$

  . If the above condition holds then CC-LPD will decode to the same point as CC-MLD.
- This is the same condition as that from the Nullspace property of the CS-LPD
- We dig a little deeper to form the connection between CC-LPD and CS-LPD

Forming the bridge

- Let H be a zero one measurement matrix. Then

$$\nu \in \textit{Nullsp}(H) \implies |\nu| \in \mathcal{K}(H)$$

Let $\omega = |\nu|$. It can be shown that ω is in the fundamental cone of H. Thus, corresponding to every ν, we have an $\omega = |\nu|$ such that ν lies in the fundamental cone of H.

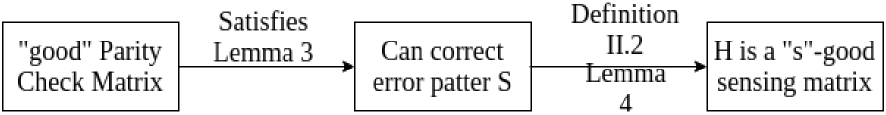
Forming the bridge

- Let H be a zero one measurement matrix. Then

$$\nu \in \ \mathit{Nullsp}(H) \implies |\nu| \in \mathcal{K}(H)$$

  Let $\omega = |\nu|$. It can be shown that ω is in the fundamental cone of H. Thus, corresponding to every ν, we have an $\omega = |\nu|$ such that ν lies in the fundamental cone of H.
- Note that the result is one way, i.e for every point in the Nullspace of H we can find a point in the conic hull of $\mathcal{K}$
- If H is a good LDPC code then any vector in the nullspace of H will satisfy the null space property provided it satisfies property(4)
- Since H is a good LDPC code, it will satisfy property (4) for a significant value of $|\mathcal{S}|$ and hence the H will act as a $|\mathcal{S}|$ good sensing matrix.

Intuition using block diagram



Interpretation

- If there is a point which is problematic for CS-LPD, then it will map to a point which will be problematic for CC-LPD.

Interpretation

- If there is a point which is problematic for CS-LPD, then it will map to a point which will be problematic for CC-LPD.
- If H is a good LDPC parity matrix, then upto some value of s we can say that there are no problematic bit-flips for CC-LPD
- Hence, there can be no point in the nullspace of H which violates the nullspace property for s sparse solutions.

Interpretation

- If there is a point which is problematic for CS-LPD, then it will map to a point which will be problematic for CC-LPD.
- If H is a good LDPC parity matrix, then upto some value of s we can say that there are no problematic bit-flips for CC-LPD
- Hence, there can be no point in the nullspace of H which violates the nullspace property for s sparse solutions.
- Hence, we can say that such a matrix is s-good.

Interpretation

- If there is a point which is problematic for CS-LPD, then it will map to a point which will be problematic for CC-LPD.
- If H is a good LDPC parity matrix, then upto some value of s we can say that there are no problematic bit-flips for CC-LPD
- Hence, there can be no point in the nullspace of H which violates the nullspace property for s sparse solutions.
- Hence, we can say that such a matrix is s-good.
- The notion of a good matrix depends on the channel being used.For a BSC channel, the performance results translate directly,i.e an s error correcting code can act as s-good matrix. For other channels, the performance translations are slightly more complicated.

Outline

Constructing the parity check matrix

- Theoretical results state that for a binary symmetric channel (BSC), if a parity-check matrix can correct any k bit-flipping errors under **CC-LPD**,then the same matrix taken as a measurement matrix over reals can be used to recover all k-sparse error signals under **CS-LPD**.
- Consider the parity check matrix of $(7,4)$ Hamming code:

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

- The parity check matrix H can correct upto 1-bit flipping error under **CC-LPD**

Recovery of s-sparse signals

- Using the theoretical result, H can be used as sensing matrix for recovering all 1-sparse vectors.

Recovery of s-sparse signals

- Using the theoretical result, H can be used as sensing matrix for recovering all 1-sparse vectors.
- For H, any $2(2s)$ columns are linearly independent and hence can be used to recover all $1(s)$-sparse vectors

Recovery of s-sparse signals

- Using the theoretical result, H can be used as sensing matrix for recovering all 1-sparse vectors.
- For H, any $2(2s)$ columns are linearly independent and hence can be used to recover all $1(s)$-sparse vectors
- Use the l1-Magic Solver in MATLAB to compute the ℓ_1-minimization estimate to **CS-LPD**

Recovery of s-sparse signals

- Using the theoretical result, H can be used as sensing matrix for recovering all 1-sparse vectors.
- For H, any $2(2s)$ columns are linearly independent and hence can be used to recover all $1(s)$-sparse vectors
- Use the l1-Magic Solver in MATLAB to compute the ℓ_1-minimization estimate to **CS-LPD**
- Consider the following 1-sparse vector x

$$x = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0.9244 \\ 0 \\ 0 \end{bmatrix}$$

Recovery of s-sparse signals

- Consider $x \mapsto Ax$, let $y = Hx$

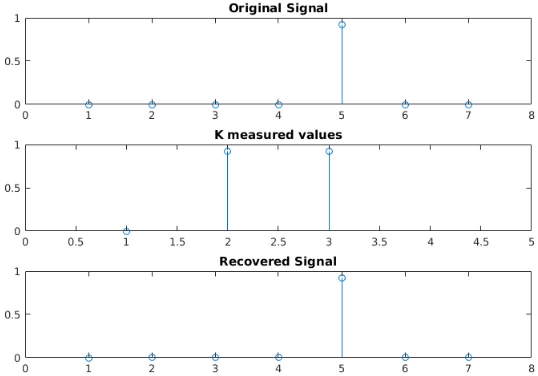
$$y = \begin{bmatrix} 0 \\ 0.9244 \\ 0.9244 \end{bmatrix}$$

- Pass y and H into the ℓ_1 minimization solver and let the estimate be $\hat{x}$

$$\hat{x} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0.9244 \\ 0 \\ 0 \end{bmatrix}$$

Verification of theoretical results

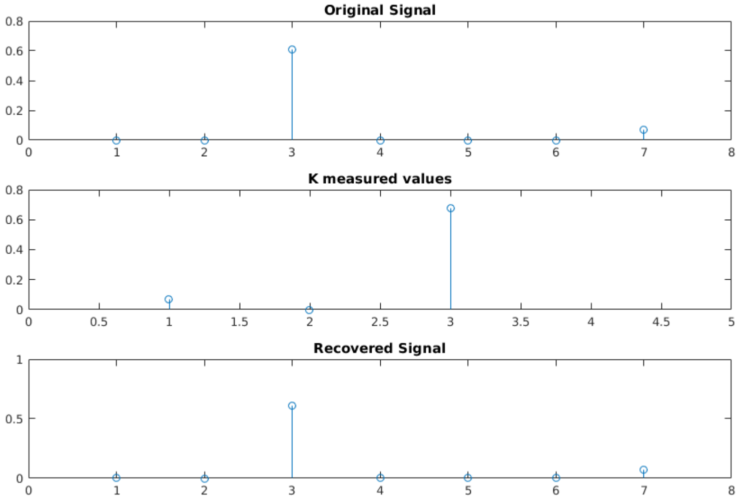
- We can see that $x = \hat{x}$.
- We are able to perfectly recover the 1-sparse signal using ℓ_1-minimization solver
- This verifies the theoretical results



Case of Approximate Recovery of 2-sparse vector

- $x = \begin{bmatrix} 0 \\ 0 \\ 0.6073 \\ 0 \\ 0 \\ 0 \\ 0.0708 \end{bmatrix}$: Original signal
- $y = \begin{bmatrix} 0.0708 \\ 0 \\ 0.6780 \end{bmatrix}$: The measurement vector
- $\hat{x} = \begin{bmatrix} 0 \\ 0 \\ 0.6071 \\ 0 \\ 0 \\ 0 \\ 0.0707 \end{bmatrix}$: Estimate of the original signal

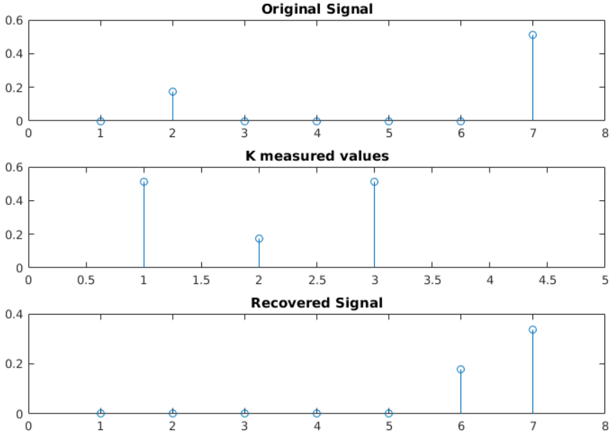
Case of Approximate Recovery of 2-sparse vector



Case of Incorrect Recovery of 2-sparse vector

- $x = \begin{bmatrix} 0 \\ 0.1767 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.5137 \end{bmatrix}$: Original Signal
- $y = \begin{bmatrix} 0.5137 \\ 0.1767 \\ 0.5137 \end{bmatrix}$: Measurement Vector
- $\hat{x} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.1766 \\ 0.3370 \end{bmatrix}$: Estimate of the original signal

Case of Incorrect Recovery of 2-sparse vector



Summary

- Established a bridge connection between **CS-LPD** and **CC-LPD**.
- Presented a deterministic polynomial time algorithm to form a zero-one sensing matrix
- Showed that a good LDPC 0-1 parity check matrix will make a good sensing matrix for BSC
- Illustrated the results with a small example
- Outlook
  - Didn't show performance guarantees translation for other channels like AWGN and BEC
  - Didn't cover the concept of M-cover of a graph and equivalent definition of fractional weights