Closed-Form Solution of Some Sparse Single-system-multiple-output (SSMO) Underdetermined Systems of Equations

Andrew E. Yagle
Department of EECS, The University of Michigan, Ann Arbor, MI 48109-2122

Abstract—**The single-system multiple-output (SSMO) sparse reconstruction problem is to solve the underdetermined linear systems of equations Y=HX where Y is M-by-L, H is M-by-N, X is N-by-L, K rows of X are not all-zero, and N>M>K>L. We show that by eliminating some variables of Y=HX we can reduce this problem into one with K=L, whose solution is trivial.**

Keywords— **Sparse reconstruction**
Phone: 734-763-9810. Fax: 734-763-1503.
Email: aey@eecs.umich.edu. **EDICS: 2-REST.**

I. INTRODUCTION

A. Problem Statement

The single-system multiple-output (SSMO) sparse reconstruction problem is to solve the multichannel underdetermined linear systems of equations

$$Y = HX \tag{1}$$

- Y is an $M \times L$ matrix having full rank.
- H is an $M \times N$ matrix having full rank.
- X is an $N \times L$ matrix having full rank.
- X has only K rows that are not all zero (NAZ). But the location of these NAZ rows is unknown.
- #channels=$L < K < M < N$.

The SSMO problem is a multichannel version of the usual sparse reconstruction problem of solving the underdetermined linear system of equations $y = Hx$ where y is an M-vector, K elements of x are nonzero, but the location of the nonzero elements of N-vector x is unknown. This is the single-system single-output (SSSO) sparse reconstruction problem.

Sparse reconstruction is currently of great interest in compressed sensing, since many real-world signals and images have sparse (mostly zero) representations in an appropriate basis, such as a set of wavelet or curvelet basis functions. The number of observations necessary to reconstruct the signal is therefore greatly reduced from the size of the signal. The SSMO problem arises in several applications in which multiple snapshots over time of a signal whose characteristics are varying over time are available.

B. Problem Background

Many approaches to solving the SSSO problem are known. The most common approach is basis pursuit, in which linear programming is used to find the solution to $y = Hx$ which has the minimum ℓ_1 norm (sum of absolute values). Another approach is matching pursuit, in which columns of H most highly correlated with the residual y–$H\hat{x}_i$ are successively chosen to minimize the residual y–$H\hat{x}_{i+1}$. Iterative thresholding, in which the Landweber iteration is applied to $y = Hx$ and some elements of $\hat{x}_i$ are thresholded to zero at each iteration. We will not attempt to summarize the many variations on these themes.

Extending these approaches to the SSMO problem has proven to be difficult. The most common approach is to find the solution to $Y = HX$ that minimizes a mixed-norm criterion such as

$$\underset{x_{ij}}{\text{MIN}} \sum_{i=1}^{N} \sqrt{\sum_{j=1}^{L} x_{ij}^2}. \tag{2}$$

This is the ℓ_1 norm over columns i of the ℓ_2 norm of each row. The ℓ_2 norm of a row is zero if and only if all elements in that row are zero, and positive otherwise. The ℓ_1 norm is then just the sum of these, and so can be expected to maximize the number of all-zero rows of X, as desired.

C. Contribution of This Paper

This paper shows that eliminating P of the rows of X results in a reduced-size problem (RSP) with:

- $\tilde{Y}$ is an $(M–P) \times L$ matrix having full rank.
- $\tilde{H}$ is an $(M–P) \times (N–P)$ matrix having full rank.
- $\tilde{X}$ is an $(N–P) \times L$ matrix having full rank.
- $\tilde{X}$ has about $K\frac{N-P}{N}$ rows that are NAZ.

The number of NAZ rows of $\tilde{X}$ is approximate and assumes that NAZ rows are randomly distributed among the $(N–P)$ rows of $\tilde{X}$.

We require $(M–P) > K\frac{N-P}{N}$ for a unique solution.

The point of the RSP is that if $K\frac{N-P}{N} \leq L$ then the RSP can be solved in closed form (see below).

II. Derivation of New Algorithm

A. Reduced-size Problem (RSP)

We partition $Y = HX$ as follows:

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \quad (3)$$

where the sizes of the various matrices are

- Y_1 is $(M{-}P) \times L$ and X_1 is $(N{-}P) \times L$.
- Y_2 and X_2 are both $P \times L$ and H_{22} is $P \times P$.
- H_{12} is $(M{-}P) \times P$ and H_{21} is $P \times (N{-}P)$.

Note that we may reorder the rows and columns of Y, H, X before partitioning the system $Y = HX$.

Eliminating X_2 gives the RSP

$$Y_1 - H_{12}H_{22}^{-1}Y_2 = (H_{11} - H_{12}H_{22}^{-1}H_{21})X_1 \quad (4)$$

which can be written as the RSP $\tilde{Y} = \tilde{H}\tilde{X}$, where

$$\begin{aligned} \tilde{Y} &= Y_1 - H_{12}H_{22}^{-1}Y_2. \\ \tilde{H} &= H_{11} - H_{12}H_{22}^{-1}H_{21}. \\ \tilde{X} &= X_1. \end{aligned} \quad (5)$$

These matrices have the sizes noted above.

B. Closed-form Solution (CFS)

If P is chosen so that $\tilde{K} = K\frac{N-P}{N} < L$ then the RSP can be solved in closed form as follows. Define

- $\{i_1, i_2 \ldots i_{\tilde{K}}\}$=NAZ rows of X.
- $\tilde{H}_{i_1 \ldots i_{\tilde{K}}}$=$(M{-}P) \times \tilde{K}$ submatrix of $\tilde{H}$.
- $X'_{i_1 \ldots i_{\tilde{K}}}$=$L \times \tilde{K}$ submatrix of X'.

Let D be the left null matrix of $\tilde{Y}$, so $D\tilde{Y}$=0. Then

$$[0] = D\tilde{Y} = D(\tilde{H}\tilde{X}) = D[\tilde{H}_{i_1 \ldots i_{\tilde{K}}}][\tilde{X}'_{i_1 \ldots i_{\tilde{K}}}]'. \quad (6)$$

But if $L \geq \tilde{K}$, then the $\tilde{K} \times L$ matrix $[X'_{i_1 \ldots i_{\tilde{K}}}]'$ has full row rank. Then the left null spaces of both Y and $[H_{i_1 \ldots i_{\tilde{K}}}]$ are identical. Hence columns $i_1 \ldots i_{\tilde{K}}$ of $Z = D\tilde{H}$ will be columns of all zeros, so that

$$[0] = D\tilde{H}_{i_1 \ldots i_{\tilde{K}}}. \quad (7)$$

All-zero columns of Z=$D\tilde{H}$ indicate NAZ rows of $\tilde{X}$.

C. Summary of New Algorithm

1. Choose integer P so $\tilde{K} < (M{-}P)$ and $\tilde{K} < L$ where $\tilde{K}$=$K\frac{N-P}{N}$.
2. Compute the RSP from the original problem.
3. Apply the CFS algorithm to the RSP.

III. Example Problem Sizes

These examples of original problem sizes

example#	variable	#1	#2	#3
#unknowns	N	100	300	400
#equations	M	75	150	150
#nonzero	K	49	74	66
#channels	L	25	50	50
Reduction	P	50	100	100

lead to these problem sizes for the RSP

example#	variable	#1	#2	#3
#unknowns	$N{-}P$	50	200	300
#equations	$M{-}P$	25	50	50
#nonzero	$\tilde{K}$	24	49	49
#channels	L	25	50	50

In all of these cases, $\tilde{Y}$ drops rank. If it does not, then choose a different set of eliminated variables X_2.

IV. NUMERICAL EXAMPLE

The following demonstrates the procedure on a 76×100, 50-sparse, 26-channel, SSMO problem.

The problem eliminates the bottom half of X; if this matrix is not 25-sparse in rows, then it switches to eliminating the top half of X, which must then be at least 25-sparse in rows, and displays a message.

The sum of absolute values of each row of the half of X used is plotted in blue, and the indicator variable Z in red; zero values indicate a NAZ row of X.

Once the locations of NAZ rows of X are found, the original problem is easily solved (is not shown).

```
clear;clf;H=rand(76,100);X=rand(100,26);
Q1=rand(1,100);Q2=sort(Q1);%random rows
K=find(Q1>Q2(50));X(K,:)=zeros(50,26);
Y=H*X; %GOAL:Find zero rows of X
HIH=H(1:26,51:100)/H(27:76,51:100);
YY=Y(1:26,:)-HIH*Y(27:76,:);
HH=H(1:26,1:50)-HIH*H(27:76,1:50);
N=(null(YY'))';X1=abs(X(1:50,:));
if(isempty(N));X1=abs(X(51:100,:));
disp('Keeping last 50 rows of X')
HIH=H(1:26,1:50)/H(27:76,1:50);
YY=Y(1:26,:)-HIH*Y(27:76,:);
HH=H(1:26,51:100)-HIH*H(27:76,51:100);
N=(null(YY'))';end;Z=sum(abs(N*HH),1);
subplot(211),stem(sum(X1,2)),hold on
subplot(211),stem(Z,'r','filled'),...
title('X in blue; indicator in red')
```

A sample run of this program is shown below.