Random Channel Coding and Blind Deconvolution

M. Salman Asif, William Mantzel and Justin Romberg

***Abstract*— Blind deconvolution arises naturally when dealing with finite multipath interference on a signal. In this paper we present a new method to protect the signals from the effects of sparse multipath channels—we modulate/encode the signal using random waveforms before transmission and estimate the channel and signal from the observations, without any prior knowledge of the channel other than that it is sparse. The problem can be articulated as follows. The original message x is encoded with an overdetermined $m \times n$ $(m > n)$ matrix A whose entries are randomly chosen; the encoded message is given by Ax. The received signal is the convolution of the encoded message with h, the s-sparse impulse response of the channel. We explore three different schemes to recover the message x and the channel h simultaneously. The first scheme recasts the problem as a block ℓ_1 optimization program. The second scheme imposes a rank-1 structure on the estimated signal. The third scheme uses nuclear norm as a proxy for rank, to recover the x and h. The simulation results are presented to demonstrate the efficiency of the random coding and proposed recovery schemes.**

I. Introduction

The problem of deconvolution naturally arises in many applications. Examples include channel equalization, image restoration and seismology. The general model for the problem can be stated in the vector form as

$$y = f * h, \tag{1}$$

where y denotes the observations of the signal f, received from a system with impulse response h. Our goal in deconvolution is typically to solve (1) for f. If we know the impulse response h, the linear deconvolution problem can be reduced to a linear inverse problem. However, if we do not know the impulse response, the deconvolution problem becomes ill-posed, and is called *blind deconvolution* [1]. In order to recover s without prior knowledge of h, we have to use some additional assumptions on the structure of s and h. These assumptions can be introduced in terms of statistical or deterministic models. For example, in blind image restoration we usually assume that images are sparse in some representation and that the blur kernel (point spread function) has compact support [2]. Blind equalization techniques in digital communication typically use the higher order statistics of the observed signal [3], [4].

In this paper we demonstrate that random coding can help us in the blind deconvolution when the unknown impulse response of the system is sufficiently sparse. This problem can be motivated with an example from digital communication, where channel is constantly varying and we cannot estimate it by sending training symbols at regular intervals. Assume that we want to transmit a signal $x \in \mathbb{R}^n$ to a remote receiver. The communication channel introduces some multipath interference to the signal, where the impulse response of channel has very few nonzero taps. In order to protect the signal from the multipath interference, we instead transmit a codeword $f = Ax$, where A is an $m \times n$ matrix with $m > n$. The entries of A are typically chosen from an i.i.d. Gaussian distribution. The received signal can be expressed as

$$y = Ax * h, \tag{2}$$

where h denotes an s-sparse channel impulse response of length L. Our goal is to recover the original signal x from the received signal y, without *a priori* knowledge of h (other than that it is sparse). We leverage the recent work in compressive sensing (CS) to guide our approach for this recovery.

Compressive sensing is generally known as a framework of estimating sparse signals from a small number of linear (incoherent) measurements [5], [6], [7], [8]. There is another class of problems, closely related to compressive sensing, where random coding is used—ℓ_1-decoding [9], [10]. In this framework we use random coding to protect an arbitrary signal from the "sparse additive" errors. The problem setup is as follows. Assume that we want to transmit a signal $x \in \mathbb{R}^n$ to the receiver, where the transmission channel can introduce a small number of errors in the transmitted signal. In order to introduce robustness against such errors, we instead transmit a codeword Ax, where A is an $m \times n$ random coding matrix with $m > n$. The received signal is

$$y = Ax + e, \tag{3}$$

where $e \in \mathbb{R}^m$ is the vector for sparse errors introduced by the channel. If e contains s nonzero terms and $m - n = O(s \log(m/s))$, then solving the following optimization program can recover x exactly (with high probability)

$$\text{minimize } \|Ax - y\|_1. \tag{4}$$

Although the random coding part in our setup is same as the one in ℓ_1-decoding, the signal recovery part requires a different set of tools. The main challenge comes from the fact that the constraints imposed by the measurements y are bilinear (and biconvex) in x and h. Assume that we want to solve the following optimization problem [11]

$$\text{minimize } \frac{1}{2}\|h * Ax - y\|_2^2 + \tau\|h\|_1 \tag{5}$$

for x and h, where the ℓ_1 norm ($\|\cdot\|_1$) imposes sparse prior on h. The biconvexity of (5) in x and h keeps us away

The authors are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA Email: {sasif, willem, jrom}@ece.gatech.edu

from solving it simultaneously for x and h. However, we can still utilize some powerful optimization tools to solve this problem. One approach is to fix some estimate x in order to solve a problem that is convex in h, and then fix this estimate for h in order to solve a linear system to solve for x, known as *alternating minimization* [11].

Another approach, and the basis for the work presented in this paper, notes that the constraints imposed by the measurements are linear in the elements $X = xh^T$, i.e. the values $x_i h_j$. Indeed, there is some linear operator $\mathcal{A}$ for which

$$\mathcal{A}(X) = y. \tag{6}$$

In the work presented here, we attempt to recover a rank-1 matrix corresponding to our transmitted signal x and our sparse channel h.

In this paper we present three recovery schemes. The first scheme formulates the recovery problem as a block sparse signal recovery problem, called block ℓ_1 scheme. The second scheme tries to impose rank-1 constraint on the solution of block ℓ_1. The third scheme minimizes the nuclear norm of the matrix $X = xh^T$ corresponding to the estimates of x and h. We also present some simulation results to demonstrate the efficacy of random coding, and the performance of our techniques.

II. Block ℓ_1

The block ℓ_1 recovery scheme utilizes a variation on traditional compressive sensing that deals with block sparse signals [12], [13]. Since the nonzero elements are clustered together, we can significantly improve the performance by utilizing this information during the recovery process. In block ℓ_1, we solve the following convex program:

$$\underset{X}{\text{minimize}} \ \|X\|_{2\to 1} = \sum_{k=1}^{L} \|X_k\|_2 \ \text{ subject to } \ \mathcal{A}(X) = y, \tag{7}$$

where X_k is the kth column of X.

We refer to $\|\cdot\|_{2\to 1}$ as the $2 \to 1$ norm. Minimizing this norm often helps to promote the column-sparsity, such that many of the columns are identically zero. In particular, if the matrix is rank 1 with $X = fg^T$, then

$$\|X\|_{2\to 1} = \sum_{k=1}^{L} \|X_k\|_2 = \sum_{k=1}^{L} |g_k| \|f\|_2 = \|f\|_2 \|g\|_1, \tag{8}$$

so that the column support of X coincides with the support of the vector g.

For our purposes, the linear operator $\mathcal{A}$ is defined via the matrix A, and maps $m \times n$ matrices to vectors as follows:

$$y = \mathcal{A}(X) = \sum_{k=1}^{L} S^k A X_k = \sum_{l=1}^{n} A_l * [X^T]_l, \tag{9}$$

where S is a shift operator and $[X^T]_l$ is the lth column of X^T (i.e. the lth row of X, transposed). In words, $\mathcal{A}$ sums the convolutions of the columns of A with the corresponding rows of X. For the sake of illustration, we often discuss $\mathcal{A}$ as a matrix acting on a vectorized matrix, though we generally do not explicitly construct this matrix in practice. Alternatively, we may express the y_k values directly as matrix inner products (using the usual trace inner product)

$$y_k = \langle S^k \tilde{A}, X^T \rangle = \mathrm{Tr}(S^k \tilde{A} X), \tag{10}$$

where $\tilde{A}$ is the time-reversed form of A to facilitate convolution.

Although this scheme improves the recovery performance as compared to the standard basis pursuit (where we use ℓ_1 norm over the complete vector) [14]. However, this gives us the desired answer only if $m \gtrsim sn \log(nL)$ [15]. Whereas, we know that the matrix X has only $n+s$ degrees of freedom. The reason for this "poor" performance by block ℓ_1 is that it only attempts to minimize the number of active columns in X (i.e., columns with nonzero norm) and does not enforce the rank-1 constraint which is a desired attribute of our solution. In next section we discuss an iterative method to enforce the rank-1 constraint on the solution of block ℓ_1 method.

III. Rank-constrained gradient descent

Ideally we would like to solve the following optimization problem with block ℓ_1

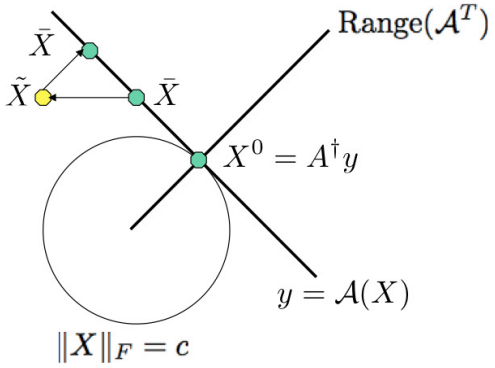
$$\begin{aligned} \text{minimize} \ & \sum_{k=1}^{L} \|X_k\|_2 \\ \text{subject to} \ & y = \mathcal{A}(X) \\ & \mathrm{rank}(X) = 1. \end{aligned}$$

Indeed, the rank constraint would critically reduce the number of degrees of freedom from $O(ns)$ to $O(n+s)$, enabling exact recovery with substantially fewer measurements. However, the rank constraint makes it a non-convex problem and solving such a program is known to be NP-hard [16].

In this section, we discuss a first order gradient descent method that enforces the rank-1 constraint on the block ℓ_1 solution, at every gradient step. This method seeks to traverse the affine space defined by the measurement constraints $\mathcal{A}(X) = y$ for the sparsest rank-1 matrix using simple gradient steps.

Recall that our observations have the linear form $y = \mathcal{A}(X)$. Note that the set of all X satisfying $y = \mathcal{A}(X)$ is defined by the affine space $H = \{X_0 + \tilde{X} : \tilde{X} \in \mathrm{Null}(\mathcal{A})\}$ for some X_0 satisfying $\mathcal{A}(X_0) = y$. As illustrated in Fig. 1, the key idea for this approach is to start with an initial estimate X_0 and make gradient steps within the affine space H towards our solution. A sensible initial estimate is $X_0 = \mathcal{A}^\dagger y = \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1} y$ which is the minimal X (in Frobenius norm) satisfying the constraints. At each iteration, we search for the nearest rank-1 matrix, take a gradient step from that matrix to minimize the $2 \to 1$ norm, and then project it back to the affine subspace H.

Next, we consider how to implicitly compute the null space projection $\tilde{X} = P(X)$ such that $\tilde{X}$ is the closest matrix to X in $\mathrm{Null}(\mathcal{A})$. Since $\mathcal{A}^\dagger\mathcal{A}$ is the projection operator onto the $\mathrm{Range}(\mathcal{A}^T)$, it follows that the projection onto the $\mathrm{Null}(\mathcal{A})$ is $P = I - P^\perp = I - \mathcal{A}^\dagger\mathcal{A}$, orthogonal complement.



(a)

Fig. 1. An illustration of trace-constrained gradient descent. The observations $y = \mathcal{A}(X)$ define an affine space that the solution X must satisfy. After choosing the initial estimate of minimal norm that lies on this affine space, at each iteration a gradient step is made followed by a reprojection onto this space.

A. Implicit Projection

Since the vector y is simply a sum of convolutions as in (9), the operator $\mathcal{A}$ has a simple relation in the Fourier domain. Indeed, the jth element of $\mathcal{A}(X)$ in the Fourier domain is just a sum of the products of corresponding elements of the Fourier versions of X and A:

$$\hat{\mathcal{A}}(\hat{X})_j = \sum_i \hat{A}_{ij}\hat{X}_{ij} = \langle \hat{A}_j, \hat{X}_j \rangle,$$

where F is the DFT matrix such that $\hat{A} = A^T F$ is the row-wise Fourier transform of A^T and $\hat{X} = XF$ is the row-wise Fourier transform of X. Since each Fourier component of $\mathcal{A}(X)$ is simply computed with an inner product involving only that column, the projection operator $\hat{P}^\perp = \hat{\mathcal{A}}^\dagger\hat{\mathcal{A}}$ decouples each column $\hat{X}_j$ so that the projection can be computed on a per-column basis in the Fourier domain. In this way, the jth column of the projection in the Fourier domain is given as a simple projection onto the normalized $\hat{A}_j$:

$$\hat{P}^\perp(\hat{X})_j = \hat{A}_j(\hat{A}_j^*\hat{A}_j)^{-1}\hat{A}_j^*\hat{X}_j = \frac{\hat{A}_j^*\hat{X}_j}{\|\hat{A}_j\|^2}\hat{A}_j$$

If P is not conducive to this block diagonalization, as may be the case when $\mathcal{A}$ operator represents linear convolution, it may be necessary to construct $\mathcal{A}$ explicitly and compute this projection via the QR decomposition of $\mathcal{A}$, generally computationally more expensive than implicit projection, but it will not be too prohibitive for medium scale problem.

Now with this matrix projection operator, we can easily compute the closest matrix $\bar{X}$ in the affine space $\mathcal{A}(X) = y$ as $\mathcal{A}^\dagger(y - \mathcal{A}(X)) = X_0 + P(X)$.

B. Gradient Descent

Assume that we have an estimate $\bar{X}$ on the affine space H. Now that we are able to reproject our current estimate $\bar{X}$ to $P\bar{X} + X_0$ in order to meet the measurement constraints, we discuss our gradient descent approach to find the sparsest rank-1 matrix that meets these constraints.

Since the reprojected matrix is not necessarily rank-1, our first step is to compute the closest rank-1 matrix $\sigma_1 U_1 V_1^*$ from the singular value decomposition $\bar{X} = U\Sigma V^*$. From this matrix, we take a gradient step to minimize $\|X\|_{2\to 1}$. In general, such a gradient step involves reducing each column proportionally to its unit vector and takes the form $\Delta X_k = -\alpha\frac{X_k}{\|X_k\|}$ when $X_k \neq 0$ and 0 otherwise for some step size α. However, in our case all vectors are scalar multiples of U_1 with $\bar{X}_k = (\sigma_1 V_{1k})U_1$, so it is equivalent to reducing their magnitudes $(\sigma_1|V_{1k}|)$ uniformly, as one would do when taking a gradient step of an ℓ_1 norm. In particular, a gradient step of $\beta = \sigma_1\alpha$ on each column $X_k = \sigma_1 V_{1k} U_1$ would produce the following estimate $\tilde{X}$:

$$\begin{aligned}
\tilde{X}_k &= X_k - \sigma_1\alpha\frac{X_k}{\|X_k\|} \\
&= (\|X_k\| - \sigma_1\alpha)\frac{X_k}{\|X_k\|} \\
&= (\sigma_1|V_{1k}| - \sigma_1\alpha)U_1\mathrm{sgn}(V_{1k}) \\
&= \sigma_1\mathrm{sgn}(V_{1k})(|V_{1k}| - \alpha)U_1 \\
&= \sigma_1\tilde{V}_{1k}U_1. \qquad (11)
\end{aligned}$$

Note that because the σ_1 constant is common to all columns, it is equivalent to consider this as a uniform reduction in $|V_{1k}|$. Also, because the gradient is zero when $X_k = 0$, a sensible implementation of this gradient descent would modify $\tilde{V}_{1k} = \mathrm{sgn}(V_{1k})(|V_{1k}| - \alpha)$ to $\tilde{V}_{1k} = \mathrm{sgn}(V_{1k})\max(|V_{1k}| - \alpha, 0)$. In other words, we use soft thresholding on V_1.

Combining these steps gives the following algorithm:

1) **Initialize** $\bar{X} = X_0$, initialize α.
2) **Compute** the SVD $\bar{X} = U\Sigma V^*$.
3) **Soft threshold** V_1 with parameter α as $\tilde{V}_1$.
4) **Reproject** $\bar{X}$ as $X_0 + P(\sigma_1 U_1 \tilde{V}_1^*)$.
5) **Update** α to decrease geometrically.
6) **Iterate** through all but the first step.

The rank-1 projection step could potentially be done directly on X or its Fourier counterpart $\hat{X}$. However, the soft thresholding step cannot be performed in Fourier space, so at least one FFT and one inverse FFT are necessary at every iteration. Power methods can be used to compute V_1 from $\bar{X}$ and it would be computationally advantageous [17].

IV. Nuclear norm minimization

In this section, we discuss nuclear norm minimization setting to estimate X from the measurements $y = \mathcal{A}(X)$. As we discussed in sec. III, one key challenge with the matrix formulation is that the rank constraint is difficult to incorporate because it is not convex. It has been shown in recent literature that the nuclear norm is the closest convex relaxation of matrix rank and it serves as a good proxy [18], [16]. The optimization problem for nuclear norm minimization can be stated as

$$\text{minimize } \|X\|_* \text{ subject to } \mathcal{A}(X) = y, \qquad (12)$$

where $\|X\|_*$ denotes the nuclear norm of X, which is defined as the sum of all the singular values of X. The equivalent semidefinite program can be written as

$$\underset{X,W_1,W_2}{\text{minimize}} \ \frac{1}{2}[\text{Tr}(W_1) + \text{Tr}(W_2)] \tag{13}$$

$$\text{subject to} \ \begin{bmatrix} W_1 & X \\ X^T & W_2 \end{bmatrix} \succeq 0$$

$$\mathcal{A}(X) = y. \tag{14}$$

The results in [16] suggest that if the mapping $\mathcal{A}$ behaves as a restricted isometry, then the solution to (12) is unique and exact. In our setting, this conveniently translates to a guaranteed recovery of x and h whenever $m > O(L+n)$, if our convolutive operator obeys some restricted isometry.

Note that when the solution of (12) (equivalently (13)) gives a rank-1 matrix, we can write

$$\bar{Z} = \begin{bmatrix} \bar{W}_1 & \bar{X} \\ \bar{X}^T & \bar{W}_2 \end{bmatrix} = \begin{bmatrix} \bar{x} \\ \bar{h} \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{h} \end{bmatrix}^T,$$

and it implies that $\text{Tr}(\bar{Z}) = \text{Tr}(W_1) + \text{Tr}(W_2) = \|\bar{x}\|_2^2 + \|\bar{h}\|_2^2$ is also minimal. This is the best we can do when h is an arbitrary vector, but less than ideal in the case of sparse h, as it does not capture our preference that $\bar{h}$ should be sparse. However, it mitigates another issue we might have had. Namely, x and h are only determined up to a scale factor so that αx and $\alpha^{-1}h$ would also suffice. Our objective resolves the ambiguity by choosing them on the same approximate scale. Still, it may be preferable to impose a norm constraint on h as

$$\text{Tr}(W_2) = 1, \tag{15}$$

to explicitly resolve the ambiguity. Now $\|\bar{h}\|_2 = 1$, and $\|\bar{x}\|_2$ is minimal. Since $\|h\|_1 \leq \sqrt{s}\|h\|_2$ whenever h is s-sparse, we may additionally place the following ℓ_1 constraint $\|h\|_1^2 \leq s$ to give preference to such solutions:

$$\mathbf{1}^T |W_2| \mathbf{1} \leq s. \tag{16}$$

The semidefinite program (13) can be solved in a variety of ways, e.g., using interior point method [19]. Although very accurate, such second order methods are computationally expensive. For large parameters, this quickly becomes infeasible, and first order gradient descent methods may be preferable [20].

V. Simulation results

To evaluate the performance of these routines and demonstrate the efficacy of the random coding, we performed simulations over a range of m (codeword length), n (signal length or information), L (channel length) and s (channel sparsity) to determine the empirical probability of success in different regions. Figures 2 and 3 plot the phase transition diagrams for rank-constrained gradient descent and nuclear norm minimization respectively. The results were computed for fixed values of m by varying the variables n, L and s that illustrate the boundaries that separate the regions when these approaches succeed and when they fail. We ran one simulation per pixel and used local weighted averages (i.e. a smoothing filter) to compute these probability estimates, under the assumption that the underlying probability field is continuous and relatively smooth.

The simulation results from rank-constrained gradient descent for both linear and circular convolution are shown in Fig. 2 for fixed m and L, and variable information n and sparsity s. As shown in Fig. 2, this approach is able to recover the original signal for sufficiently sparse channels. Comparing the Fig. 2b to the left half of Fig. 2c shows that shorter channel impulse response tends to outperform longer impulse response channels for comparable information rates and sparsity. This is somewhat intuitive because the support of the channel is known to a much greater degree for shorter channels. Also, the sparsity tends to play a bigger role for larger channels and recovery is much less certain for large channels that are not sufficiently sparse.

The simulation results for rank minimization scheme with linear convolution model as shown in Fig. 3 for fixed m and L, and variable information n. The results show that nuclear norm minimization successfully recovers the underlying rank-1 matrix for a range of values for n and L. Figure 3b shows slight improvement in the recovery for sparse channels, but the effect is very minor as we expect that the nuclear norm is indifferent to the sparsity of the matrix.

VI. Conclusion

We proposed a random coding scheme to protect an arbitrary signal from the effects of a sparse channel. The desired signal can be recovered from the received signal, given the channel impulse response is sufficiently sparse. The simulation results demonstrate that the redundancy in the coding, required for exact recovery, scales with the sparsity of the channel and length of the original signal.

References

[1] S. Haykin, *Adaptive filter theory*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1996.

[2] T. Chan and C. Wong, "Total variation blind deconvolution," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, 1998.

[3] L. Tong, G. Xu, T. Kailath, *et al.*, "Blind identification and equalization based on second-order statistics: A time domain approach," *IEEE Transactions on Information Theory*, vol. 40, no. 2, pp. 340–349, 1994.

[4] H. Liu, G. Xu, L. Tong, and T. Kailath, "Recent developments in blind channel equalization: From cyclostationarity to subspaces," *Signal Processing*, vol. 50, no. 1-2, pp. 83–99, 1996.

[5] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

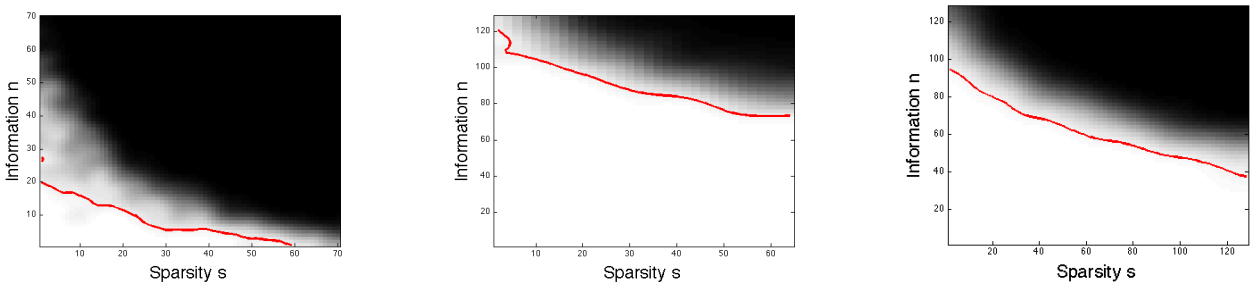
[6] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *Information Theory, IEEE Transactions on*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[7] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math*, vol. 59, no. 8, pp. 1207–1223, 2006.

[8] D. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, April 2006.

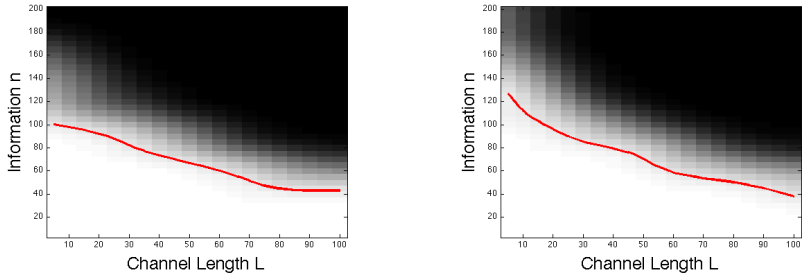
[9] E. Candès and T. Tao, "Decoding by linear programming," *Information Theory, IEEE Transactions on*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[10] E. J. Candès and P. A. Randall, "Highly robust error correction by convex programming," *Information Theory, IEEE Transactions on*, vol. 54, no. 7, pp. 2829–2840, 2008.



(a) (b) (c)

Fig. 2. The phase transition diagrams over n and s (for fixed m and L) for rank-constrained gradient descent. The intensity of the plots illustrates the regions of operation, bright regions correspond to high probability of success. Each line shown in red marks the empirical 95% success rate contour, within which recovery of the signal is most likely. Circular convolution for $m = 128$ is shown in (a), linear convolution for $m = 256$ is shown with $L = 64$ in (b), and $L = 128$ in (c).



(a) (b)

Fig. 3. The phase transition diagrams for nuclear norm minimization scheme using linear convolution channels with $m = 300$. The intensity of the plots illustrates the regions of operation, bright regions correspond to high probability of success. Each line shown in red marks the empirical 95% success rate contour. These diagrams range over the information n and channel length L. Results for completely dense channels in (a). For (b) 25% of channel coefficients are nonzero.

[11] M. S. Asif, W. Mantzel, and J. Romberg, "Channel Protection: Random coding meets sparse channels," *Information theory workshop*, October 2009. To appear.

[12] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," *preprint*, 2008.

[13] M. Stojnic, F. Parvaresh, and B. Hassibi, "On the reconstruction of block-sparse signals with an optimal number of measurements," *preprint*, 2008.

[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1999.

[15] J. Romberg and R. Neelamani, "Sparse channel separation using random probes," *Preprint*, 2009.

[16] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *Preprint*, 2007.

[17] G. Golub and C. Van Loan, *Matrix Computations*. Johns Hopkins University Press, 1996.

[18] M. Fazel, *Matrix rank minimization with applications*. PhD thesis, Elec. Eng. Dept, Stanford University, March 2002.

[19] M. Grant, S. Boyd, and Y.Ye., "CVX: Matlab software for disciplined convex programming." http://stanford.edu/ boyd/cvx.

[20] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Mathematical Programming, Series A*.