Fast Sparse Superposition Codes Have Near Exponential Error Probability for $R < \mathcal{C}$

Antony Joseph, *Student Member, IEEE*, and Andrew R. Barron, *Fellow, IEEE*

***Abstract*— For the additive white Gaussian noise channel with average codeword power constraint, sparse superposition codes are developed. These codes are based on the statistical high-dimensional regression framework. In a previous paper, we investigated decoding using the optimal maximum-likelihood decoding scheme. Here, a fast decoding algorithm, called the adaptive successive decoder, is developed. For any rate R less than the capacity $\mathcal{C}$, communication is shown to be reliable with nearly exponentially small error probability. Specifically, for blocklength n, it is shown that the error probability is exponentially small in $n/\log n$.**

***Index Terms*— Gaussian channel, subset selection, compressed sensing, multiuser detection, orthogonal matching pursuit, greedy algorithms, successive cancelation decoding, error exponents, achieving channel capacity.**

I. Introduction

THE ADDITIVE white Gaussian noise channel is basic to Shannon theory and underlies practical communication models. Sparse superposition codes for this channel were developed in [27], where reliability bounds for the optimal maximum-likelihood decoding were given. The present work provides a scheme, based on an adaptive decoder, with performance bounds that are comparable to the maximum-likelihood decoder.

In the familiar communication setup, an encoder maps length K input bit strings $u = (u_1, u_2, \ldots, u_K)$ into codewords, which are length n strings of real numbers $c_1, c_2, \ldots, c_n$, with power $(1/n)\sum_{i=1}^{n} c_i^2$. After transmission through the Gaussian channel, the received string $Y = (Y_1, Y_2, \ldots, Y_n)$ is modeled by,

$$Y_i = c_i + \epsilon_i \quad \text{for } i = 1, \ldots, n,$$

where the ϵ_i are i.i.d. Normal$(0, \sigma^2)$. The decoder produces an estimates $\hat{u}$ of the input string u, using knowledge of the received string Y and the codebook. The decoder makes a block error if $\hat{u} \neq u$. The reliability requirement is that, with sufficiently large n, the block error probability is small, when averaged over input strings u as well as the distribution of Y. The communication rate $R = K/n$ is the ratio of the number of message bits to the number of uses of the channel required to communicate them.

The supremum of reliable rates of communication is the channel capacity $\mathcal{C} = (1/2)\log_2(1 + P/\sigma^2)$, by traditional information theory [17], [36]. Here P expresses a control on the codeword power.

For practical coding the challenge is to achieve arbitrary rates below the capacity, while guaranteeing reliable decoding in manageable computation time.

The Gaussian channel coding problem is regarded as relevant to myriad settings involving transmission over wires or cables for internet, television, or telephone communications or in wireless radio, TV, phone, satellite or other space communications. In the next subsection we describe the framework of our codes.

A. Sparse Superposition Codes

The framework here is as introduced in [27], but for clarity we describe it again in brief. The story begins with a list $X_1, X_2, \ldots, X_N$ of vectors, each with n coordinates, which can be thought of as organized into a *design*, or *dictionary*, matrix X, where,

$$X_{n\times N} = [X_1 : X_2 : \ldots : X_N].$$

The entries of X are drawn i.i.d. Normal(0, 1). The codeword vectors take the form of particular linear combinations of columns of the design matrix.

More specifically, we assume $N = LM$, with L and M positive integers, and the design matrix X is split into L sections, each of size M. The codewords are of the form $X\beta$, where each $\beta \in \mathbb{R}^N$ belongs to the set

$$\mathcal{B} = \{\beta : \beta \text{ has exactly one non-zero in each section, with value in section } \ell \text{ equal to } \sqrt{P_{(\ell)}}\}.$$

This is depicted in figure 1. The values $P_{(\ell)}$, for $\ell = 1, \ldots, L$, chosen beforehand, are positive and satisfy

$$\sum_{\ell}^{L} P_{(\ell)} = P, \tag{1}$$

where P is the target power for our code.

The received vector is in accordance with the statistical linear model $Y = X\beta + \varepsilon$, where ε is the noise vector distributed N$(0, \sigma^2 I)$.

Accordingly, with the $P_{(\ell)}$ chosen to satisfy (1), we have $\|\beta\|^2 = P$ and hence, $\mathbb{E}\|X\beta\|^2/n = P$, for each β in $\mathcal{B}$.

Manuscript received July 7, 2012; revised June 24, 2013; accepted August 29, 2013. Date of publication November 7, 2013; date of current version January 15, 2014.

A. Joseph is with the Department of Statistics, University of California, Berkeley, CA 94709 USA (e-mail: antonyjoseph@lbl.gov).

A. R. Barron is with the Department of Statistics, Yale University, New Haven, CT 06520 USA (e-mail: andrew.barron@yale.edu).

Communicated by E. Arıkan, Associate Editor for Coding Theory.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2013.2289865

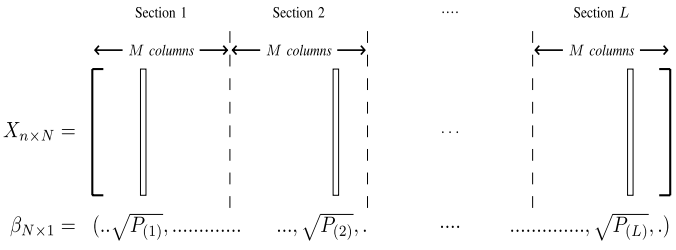

Fig. 1. Schematic rendering of the dictionary matrix X and coefficient vector β. Each vertical bar in the X matrix indicates a selected column from a section.

Here $\|.\|$ denotes the usual Euclidian norm. Thus the expected codeword power is controlled to be equal to P. Moreover, most of the codewords have power near P and the average power across the M^L codewords, given by,

$$\frac{1}{M^L}\sum_{\beta\in\mathcal{B}} \|X\beta\|^2/n$$

is concentrated at P.

Here, we consider the case of constant power allocation, where each $P_{(\ell)}$ is equal to P/L, and a variable power allocation

$$P_{(\ell)} \propto e^{-2\mathcal{C}\ell/L}, \tag{2}$$

for sections ℓ from 1 to L. These variable power allocations facilitate our adaptive successive decoder in getting the rate up to capacity. This is a slight difference from the setup in [27], where the constant power allocation was sufficient.

To explain the variable power allocation, take note of the familiar fact from multi-user Gaussian channels that the capacity of the channel $\mathcal{C} = (1/2)\log(1+P/\sigma^2)$ may be written as the sum across the sections of $(1/2)\log(1+SNIR_\ell)$, where

$$SNIR_\ell = P_\ell/(\sigma^2 + P_{\ell+1} + \cdots + P_L)$$

is the 'signal-to-(noise + interference)' level for section ℓ, if the sections were precisely decoded successively by successive interference cancellation. The exponentially decaying power allocation is the one that equalizes these section capacities to be the same, that is, equal to $\mathcal{C}/L$. As we shall see, when the parameters L and M are adjusted to have a moderate size dictionary, reliable traditional successive decoding is problematic. Consequently, we formulate an *adaptive successive decoder*, which may also be called an adaptive successive interference cancellation decoder, with desirable properties.

For ease in encoding, it is most convenient that the section size M is a power of two. Then an input bit string u of length $K = L\log_2 M$ splits into L substrings of size $\log_2 M$ and the encoder becomes trivial. Each substring of u gives the index (or memory address) of the term to be sent from the corresponding section.

As we have said, the rate of the code is $R = K/n$ input bits per channel uses and we arrange for arbitrary R less than $\mathcal{C}$. For the partitioned superposition code this rate is

$$R = \frac{L\log M}{n}.$$

For specified L, M and R, the codelength $n = (L/R)\log M$.

Control of the dictionary size is critical to computationally advantageous coding and decoding. At one extreme, L is a constant, and section size $M = 2^{nR/L}$. However, its size, which is exponential in n, is impractically large. At the other extreme $L = nR$ and $M = 2$. However, in this case the number of non-zeroes of β proves to be too dense to permit reliable recovery at rates all the way up to capacity. This can be inferred from recent converse results on information-theoretic limits of subset recovery in regression (see for [2], [43]).

Our codes lie in between these extremes. We allow L to agree with the blocklength n to within a log factor, with M arranged to be polynomial in n or L. For example, we may let $M = n$, in which case $L = nR/\log n$, or we may set $M = L$, making $n = (L\log L)/R$. For the decoder we develop here, at rates below capacity, the error probability is also shown to be exponentially small in L.

Our theory for the feasible decoder demonstrates that the decoder is able to correctly identify the terms selected for most of the sections with high probability. An outer Reed-Solomon code, operating at rate R_{outer} near 1, completes the task of correcting the small amount of mistakes made (see Subsection I-B for details). The total rate, that is, after composition with the outer code, is denoted by $R_{tot} = R_{outer}R$.

Our main result can be summarized in the following proposition. In the proposition below, we assume that the power allocation is given by (2). We denote the rate drop as,

$$\Delta = \frac{\mathcal{C} - R_{tot}}{\mathcal{C}},$$

where R_{tot} is the total rate.

In the proposition below, and the rest of this paper, we use the following notation: For quantities a, b depending on L, M or n, we denote $a = O(b)$, if $a \le \mathsf{const}\, b$, where const is a positive quantity that only depends on the signal-to-noise ratio for large n (or L or M). Similarly, $a = \Omega(b)$, if $a \ge \mathsf{const}\, b$.

Proposition 1: Let $\Delta > 0$ be fixed. Further, choose section size $M > M_0$, where $M_0 = e^{\mathsf{const}_1/\Delta^2}$. Here M can potentially depend on n. Then the adaptive successive decoder, after composition with outer Reed-Solomon code, has block error probability that is upper bounded by

$$\exp\left\{-\mathsf{const}_2 (n/\log M)\Delta^2\right\},$$

with overall decoding complexity that is $O(n^2 M)$. Here const_1, const_2 are positive constants that can be determined from the more general Theorem 2 in Subsection I-D.

Notice that from the above proposition, one sees that one can attain nearly exponentially small error probability in n, with decoding complexity that is polynomial in n. One of the drawbacks is that the section size has an exponential dependence on $1/\Delta^2$. (In [8] and [25] it is shown that this exponential dependence can be improved to $1/\Delta$ using the modified power allocation (5).) Hence the code is fast for any fixed rate drop Δ from capacity. However, for a sequence of rates approaching capacity, the corresponding requisite sizes for M makes the decoder impractical.

We also remark that this dependence is not intrinsic to the method of code construction as seen in the analysis of the optimal decoder [27] (see subsection I-E). Here optimal

decoding for minimal average probability of error consists of finding the codeword $X\beta$, with coefficient vector $\beta \in \mathcal{B}$, that maximizes the posterior probability, conditioned on X and Y. This coincides, in the case of equal prior probabilities, with the maximum likelihood rule of seeking

$$\arg\min_{\beta \in \mathcal{B}} \|Y - X\beta\|.$$

Performance bounds for such optimal, though computationally infeasible, decoding are developed in the companion paper [27]. Since no dependence between M and Δ is required there, it is an open question whether such a result can be also obtained using a feasible scheme.

B. Intuition Behind the Algorithm

The algorithm we analyze is similar in spirit to iterative algorithms like the relaxed greedy algorithm [24], forward stepwise regression [30], and Orthogonal Matching Pursuit [31]. Below, we provide a high-level description of the algorithm. Section II details the modifications we make to this algorithm so as to facilitate analysis.

Denoting as $J = \{1, 2, \ldots, N\}$ the set of terms required to be decoded. For step $k = 1$, do the following

- For $j \in J$, compute the normalized inner product of X_j with the received string Y, given by,

$$\mathcal{Z}_{1,j} = \frac{X_j^T Y}{\|Y\|},$$

- Update dec_1, the set of terms detected in the first step, as

$$\mathsf{dec}_1 = \{j \in J : \mathcal{Z}_{1,j} \geq \tau\}.$$

  Here τ is a positive threshold value.

This completes the first step of the algorithm.

Denote $P_j = P_{(\ell)}$ if j is in section ℓ. Also denote the set of terms decoded till the k-th step as

$$\mathsf{Dec}_k = \mathsf{dec}_1 \cup \mathsf{dec}_2 \ldots \cup \mathsf{dec}_k.$$

Next, perform the following for steps $k \geq 2$, and k at most a predefined number m.

- Compute the fit vector for the $k-1$-th step given by,

$$F_{k-1} = \sum_{j \in \mathsf{Dec}_{k-1}} \sqrt{P_j}\, X_j,$$

  along with the associated residual vector,

$$R_k = Y - F_{k-1}.$$

- Denoting

$$J_k = J - \mathsf{Dec}_{k-1},$$

  that is terms not decoded previously, calculate

$$\mathcal{Z}_{k,j}^{res} = \frac{X_j^T R_k}{\|R_k\|}, \qquad \text{for each } j \in J_k.$$

- Update

$$\mathsf{dec}_k = \{j \in J_k : \mathcal{Z}_{k,j}^{res} \geq \tau\}.$$

- This completes the k-th step. Stop if either L terms have been decoded, or if dec_k is empty, or if $k = m$. Otherwise increase k by 1 and repeat the above steps.

Ideally, the decoder selects one term from each section, producing an output which is the index of the selected term. For a particular section, there are three possible ways a mistake could occur when the algorithm is completed. The first is an *error*, in which the algorithm selects exactly one wrong term in that section. The second case is when two or more terms are selected, and the third is when no term is selected. We call the second and third cases *erasures* since we know for sure that in these cases an error has occurred. Let $\hat{\delta}_{mis,error}$, $\hat{\delta}_{mis,erasure}$ denote the fraction of sections with error, erasures respectively. Denote the section mistake rate,

$$\hat{\delta}_{mis} = 2\,\hat{\delta}_{mis,error} + \hat{\delta}_{mis,erase}, \tag{3}$$

where the subscript mis stands for mistakes. Our analysis provides a good bound, denoted by δ_{mis}, on $\hat{\delta}_{mis}$ that is satisfied with high probability.

The outer Reed-Solomon codes completes the task of identifying the fraction of sections that have errors or erasures (see section VI of Joseph and Barron [27] for details) so that we end up with a small block error probability. If $R_{outer} = 1 - \delta$ is the rate of an RS code, with $0 < \delta < 1$, then a section mistake rate $\hat{\delta}_{mis}$ less than δ_{mis} can be corrected, provided $\delta_{mis} < \delta$. Further, if R is the rate associated with our inner (superposition) code, then the total rate after correcting for the remaining mistakes is given by $R_{tot} = R_{outer} R$. The end result, using our theory for the distribution of the fraction of mistakes of the superposition code, is that the block error probability is exponentially small in $n/\log M$. One may regard the composite code as a superposition code in which the subsets are forced to maintain at least a certain minimal separation, so that decoding to within a certain distance from the true subset implies exact decoding.

C. Analysis

The algorithm we analyze, although very similar in spirit, is a modification of the above algorithm. These modifications are mainly in the definition of $\mathcal{Z}_{k,j}^{res}$, for $k \geq 2$. The modified version of $\mathcal{Z}_{k,j}^{res}$ is called $\mathcal{Z}_{k,j}^{comb}$, where the superscript $comb$ stands for 'combined' since this statistic is a linear combination of other statistics. Section II describes these modifications in detail.

Denote as

$$\mathsf{sent} = \{j : \beta_j \neq 0\} \qquad \text{and} \qquad \mathsf{other} = \{j : \beta_j = 0\}.$$

The set sent consists of one term from each section, and denotes the set of correct terms, while other denotes the set of wrong terms. Our analysis demonstrates that $\mathcal{Z}_{k,j}^{comb}$ is approximately a shifted normal, that is

$$\mathcal{Z}_{k,j}^{comb} \approx \mathsf{shift}_{k,j} 1_{\{j \in \mathsf{sent}\}} + N_{k,j}.$$

Here $1_{\{j \in \mathsf{sent}\}}$ is the indicator of the set $\{j \in \mathsf{sent}\}$. Further, $N_{k,j}$ is a normal random variable with mean zero and variance

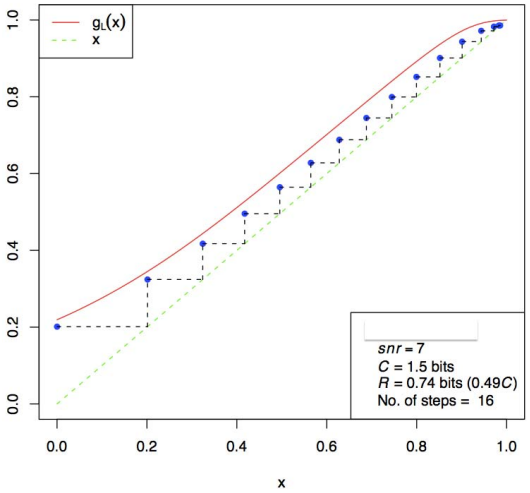

Fig. 2. Plot of the update function $g_L(x)$. The dots measure the proportion of sections correctly detected after a particular number of steps. Here $M = 2^{16}$, $\mathsf{snr} = 7$, $R = 0.74$ and L taken to be equal to M. The height reached by the $g_L(x)$ curve at the final step corresponds to a 0.986 proportion of section correctly detected, and a failed detection rate target of 0.013. The accumulated false alarm rate bound is 0.008. The probability of mistake rates larger than these targets is bounded by 1.5×10^{-3}.

near one, and $\mathsf{shift}_{k,j}$ is a positive quantity that takes the form,

$$\mathsf{shift}_{k,j} = \sqrt{\frac{C_{j,R,h}}{1 - x_{k-1}v}}.$$

Here $C_{j,R,h}$, to be defined later on, is a positive quantity that depends on the power allocation P_j, noise variance σ^2, and blocklength n. Further $v = P/(P + \sigma^2)$. The quantity x_{k-1} is closely linked to the fraction of correct detections and false alarms among terms detected till step $k - 1$. The quantity $\mathsf{shift}_{k,J}$ is the square root of a signal-to-(noise + interference) ratio

$$nP_\ell/[\sigma^2 + P(1 - x_{k-1})],$$

where the interference of the adaptive successive decoder is $P(1 - x_{k-1})$, which comes from a sum of interference contributions across the sections.

Notice that the mean shift in the distribution of $\mathcal{Z}_{k,j}^{comb}$ is non-zero only for $j \in \mathsf{sent}$. The greater this mean shift, the better is the chance of detecting the terms in sent from those in other. In particular, denoting

$$\mu_j(x) = \sqrt{\frac{C_{j,R,h}}{1 - xv}} - \tau,$$

the analysis leads us to the function $g_L : [0, 1] \to [0, 1]$, called the *success rate update function*, given by

$$g_L(x) = \sum_{j \in \mathsf{sent}} \pi_j \bar{\Phi}(-\mu_j(x)), \tag{4}$$

where $\bar{\Phi} = 1 - \Phi$, with Φ being the normal cumulative distribution function. Also, $\pi_j = P_j/P$. If x is the previous success rate, then $g_L(x)$ quantifies the expected success rate after the next step. An example of the role of g_L is shown in figure 2.

Our iterative distributional analysis has parallels with recent work on approximate message passing algorithms for compressed sensing problems [11], [12]. Indeed, our iterative characterization using the function g_L can be thought as the equivalent of the state evolution iterations discussed in these works. A distinction though is that our distributional characterizations are non-asymptotic in nature.

D. Performance of the Algorithm

Here we describe the effect of various power allocations. Further, we also provide in Theorem 2 a description of the rates achieved, along with the associated error probabilities. We also discuss the decoding complexity.

With constant power allocation, that is with $P_{(\ell)} = P/L$ for each ℓ, the decoder is shown to reliably achieve rates up to a threshold rate $R_0 = (1/2)P/(P + \sigma^2)$, which is less than capacity. This rate R_0 is seen to be close to the capacity when the signal-to-noise ratio snr is low. However, since it is bounded by $1/2$, it is substantially less than the capacity for larger snr. To bring the rate higher, up to capacity, we use variable power allocation with power given by (2).

As we have suggested, the variable power allocation (2) would arise if one were attempting to successively decode one section at a time, with the signal contributions of as yet un-decoded sections treated as noise, in a way that splits the rate $\mathcal{C}$ into L pieces each of size $\mathcal{C}/L$; however, such decoding would require the section sizes to be exponentially large to achieve desired reliability. In contrast, in our adaptive scheme, many of the sections are considered each step.

For rate near capacity, it helpful to use a modified power allocation, where

$$P_{(\ell)} \propto \max\{e^{-2\mathcal{C}\frac{\ell-1}{L}}, \mathsf{cut}\}, \tag{5}$$

with a non-negative value of cut. However, since its analysis is more involved we do not pursue this here. Interested readers may refer to documents [8], [25] for a more thorough analysis including this power allocation.

Proposition 1 follows from the following more general theorem. Once again, we assume that the power allocation is given by (2). Our choice for the value of the threshold τ, and the maximum number of steps m, will be specified later on.

We allow rate R up to $\mathcal{C}^*$, where $\mathcal{C}^*$ can be written as

$$\mathcal{C}^* = \frac{\mathcal{C}}{1 + drop^*}.$$

Here $drop^*$ is a positive quantity given explicitly later in this paper. It is near

$$\delta_M = \frac{1}{\sqrt{\pi \log M}}, \tag{6}$$

ignoring terms of smaller order.

Thus $\mathcal{C}^*$ is within order $1/\sqrt{\log M}$ of capacity and tends to $\mathcal{C}$ for large M. With the modified power allocation (5), it is shown in [8] and [25] that one can make $\mathcal{C}^*$ within order $1/\log M$ of capacity.

Theorem 2: For any inner code rate $R < \mathcal{C}^*$, express it in the form

$$R = \frac{\mathcal{C}^*}{1 + \kappa/\log M}, \tag{7}$$

with $\kappa \geq 0$. Then, for the partitioned superposition code,

I) The adaptive successive decoder admits fraction of section mistakes less than

$$\delta_{mis} = \frac{3\kappa + 5}{8\mathcal{C} \log M} + \frac{\delta_M}{2\mathcal{C}} \tag{8}$$

except in a set of probability not more than

$$p_e = \kappa_{1,M} e^{-\kappa_2 L \min\{\kappa_3 (\Delta^*)^2, \kappa_4 (\Delta^*)\}},$$

where

$$\Delta^* = (\mathcal{C}^* - R)/\mathcal{C}^*.$$

Here $\kappa_{1,M}$ is a constant to be specified later that is only polynomial in M. Also, κ_2, κ_3 and κ_4 are constants that depend on the snr. See subsection VII-E for details.

II) After composition with an outer Reed Solomon code the decoder admits block error probability less than p_e, with the composite rate being $R_{tot} = (1 - \delta_{mis})\, R$.

The proof of the above theorem is given in subsection VII-E.

Proposition 1 follows from Theorem 2 since if κ is of order $\sqrt{\log M}$, then Δ^* is of order $1/\sqrt{\log M}$. Further, as $\mathcal{C}^*$ is of order $1/\sqrt{\log M}$ below the capacity $\mathcal{C}$, we also get that $\Delta_{inner} = (\mathcal{C} - R)/\mathcal{C}$ is also $1/\sqrt{\log M}$ below capacity. From the expression for δ_{mis} in (8), one sees that the same holds for the total rate drop, that is $\Delta = (\mathcal{C} - R_{tot})/\mathcal{C}$. Correspondingly, M, or equivalently n, needs to be $e^{\Omega(1/\Delta^2)}$. A more rigorous proof is given in subsection VII-E.

Concerning the decoding complexity, as discussed in section III, an important feature of the adaptive successive decoder is that it can be arranged that the order $n^2 M$ refers to the total work space of the decoder and not the decoding time. The inner product steps can be pipelined and parallelized so that the inner code decoding time is $O(1)$ per received symbol, and it reliably incurs at most a small fraction of mistakes. Decoding time of $O(1)$ per received symbol is necessary for practical positive rate communication. It allows an achieved positive rate defined here as a ratio of number of information bits to the number of symbols received to appropriately correspond to a non-vanishing rate defined for practical purposes as the number of information bits decoded per second. The decoding complexity of the outer code is of similar order (not more than $n^2 M$), though it is unknown to us whether one can similarly trade off space and time complexity with Reed Solomon decoders to make the composite decoder time $O(1)$ per received symbol.

E. Comparison With Least Squares Estimator

Here we compare the rate achieved here by our practical decoder with what is achieved with the theoretically optimal, but possibly impractical, least squares decoding of these sparse superposition codes shown in the companion paper [27].

With power allocated equally across sections, that is with $P_{(\ell)} = P/L$, it was shown in [27] that for any $\delta_{mis} \in [0, 1)$, the probability of more than a fraction δ_{mis} of mistakes, with least squares decoding, is less than

$$\exp\{-nc_1 \min\{\Delta^2, \delta_{mis}\}\},$$

for any positive rate drop Δ and any size n. Here c_1 is a positive constant that depends only on the signal-to-noise ratio. The dependence on blocklength and rate drop in the above error exponent is similar to that of the theoretically best possible error exponent for any decoding scheme, as established by Shannon and Gallager, and reviewed for instance in [32].

The bound obtained for the least squares decoder is better than that obtained for our practical decoder in its freedom of any choice of mistake fraction, rate drop and size of the dictionary matrix X. Here, we allow for rate drop Δ to be of order $1/\sqrt{\log M}$. Further, from the expression (8), we have δ_{mis} is of order $1/\sqrt{\log M}$, when κ is taken to be of $O(\sqrt{\log M})$. Consequently, we compare the error exponents obtained here with that of the least squares estimator of [27], when both Δ and δ_{mis} are of order $1/\sqrt{\log M}$.

Using the expression given above for the least squares decoder one sees that the exponent is of order $n/(\log M)$, or equivalently L, using $n = (L \log M)/R$. For our decoder, the error probability bound is seen to be exponentially small in $L/(\log M)$ using the expression given in Theorem 2. This bound is within a $(\log M)$ factor of what we obtained for the optimal least squares decoding of sparse superposition codes.

F. Related Work in Coding

We point out several directions of past work that connect to what is developed here. The analysis of concatenated codes Forney [21] is an important forerunner to the development of code composition we give here. For the theory, he paired an outer Reed-Solomon code with concatenation of optimal inner codes of Shannon-Gallager type, while, for practice, he paired such an outer Reed-Solomon code with binary inner codes based on linear combinations of orthogonal terms (for target rates less than 1 such a basis is available), in which all binary coefficient sequences are possible codewords.

Modern day communication schemes, for example LDPC [22] and Turbo Codes [13], have been demonstrated to have empirically good performance. Both LDPC and our codes take advantage of sparsity. The former uses a sparse parity check matrix and a full set of coefficient vectors, whereas we use a full design matrix and a sparse set of coefficient vectors.

These LDPC and Turbo codes use message passing algorithms for their decoding. Interestingly, there has been recent work by Bayati and Montanari [12] that has extended the use of these algorithms for estimation in the general high-dimensional regression setup with Gaussian X matrices. Unlike our adaptive successive decoder, where we decide whether or not to select a particular term in a step, these iterative algorithms make soft decisions in each step. However, analysis addressing rates of communication have not been given in these works. Subsequent to the present work, an alternative algorithm with soft-decision decoding for our partitioned superposition codes is proposed and analyzed by Barron and Cho [6].

A different approach to reliable and computationally-feasible decoding is in the work on *channel polarization* of [3], [4]. These polar codes have been adapted to the Gaussian

case as in [1], however, the error probability is exponentially small in $\sqrt{n}$, rather than n.

The ideas of *superposition codes*, *rate splitting*, and *successive decoding* for Gaussian noise channels began with Cover [16] in the context of multiple-user channels. There, each section corresponds to the codebook for a particular user, and what is sent is a sum of codewords, one from each user. Here we are putting that idea to use for the original Shannon single-user problem, with the difference that we allow the number of sections to grow with blocklength n, allowing for manageable dictionaries. Parallel to ours, similar techniques have been used in the analysis on on-off random access channels in [20].

Other developments on broadcast channels by Cover [16], that we use, is that for such Gaussian channels, the power allocation can be arranged as in (2) such that messages can be peeled off one at a time by successive decoding. However, such successive decoding applied to our setting would not result in the exponentially small error probability that we seek for manageable dictionaries. It is for this reason that instead of selecting the terms one at a time, we select multiple terms in a step adaptively, depending upon whether their correlation is high or not.

A variant of our regression setup was proposed by Tropp [39] for communication in the single user setup. However, his approach does not lead to communication at positive rates, as discussed in the next subsection.

There have been recent works that have used our partitioned coding setup for providing a practical solution to the Gaussian source coding problem, as in Kontoyiannis et al. [28] and Venkataramanan et al. [42]. A successive decoding algorithm for this problem is being analyzed by Venkataramanan et al. [41]. An intriguing aspect of the analysis in [42] is that the source coding proceeds successively, without the need for adaptation across multiple sections as needed here.

G. Relationships to Sparse Signal Recovery

Here we comment on the relationships to high-dimensional regression. A very common assumption is that the coefficient vector is *sparse*, meaning that it has only a few, in our case L, non-zeroes, with L typically much smaller than the dimension N. Note, unlike our communication setting, it is not assumed that the magnitude of the non-zeroes be known. Most relevant to our setting are works on *support recovery*, or the recovery of the non-zeroes of β, when β is typically allowed to belong to a set with L non-zeroes, with the magnitude of the non-zeroes being at least a certain positive value.

Popular techniques for such problems involve relaxation with an ℓ_1-penalty on the coefficient vector, for example in the basis pursuit [15] and Lasso [37] algorithms. An alternative is to perform a smaller number of iterations, such as we do here, aimed at determining the target subset. Such works on sparse approximation and term selection concerns a class of iterative procedures which may be called relaxed greedy algorithms (including orthogonal matching pursuit or OMP) as studied in [5], [10], [23], [24], [26], [29], [31], [40], and [46]. In essence, each step of these algorithms finds, for a given set of vectors, the one which maximizes the inner product with the residuals from the previous iteration and then uses it to update the linear combination. Our adaptive successive decoder is similar in spirit to these algorithms.

Results on support recovery can broadly be divided into two categories. The first involves determining, for a given X matrix, uniform guarantees for support recovery. In other words, it guarantees, for any β in the allowed set of coefficient vectors, that the probability of recovery is high. The second category of research involves results where the probability of recovery is obtained after certain averaging, where the averaging is over a distribution of the X matrix.

For the first approach, a common condition on the X matrix is the *mutual incoherence condition*, which assumes that the correlation between any two distinct columns be small. In particular, assuming that $\|X_j\|^2 = n$, for each $j = 1, \ldots, N$, it is assumed that,

$$\frac{1}{n} \max_{j \neq j'} \left| X_j^{\mathrm{T}} X_{j'} \right| \quad \text{is } O(1/L). \tag{9}$$

Another related criterion is the *irrepresentable criterion* [38], [47]. However, the above conditions are too stringent for our purpose of communicating at rates up to capacity. Indeed, for i.i.d Normal(0, 1) designs, n needs to be $\Omega(L^2 \log M)$ for these conditions to be satisfied. Here $n = \Omega(L^2 \log M)$ denotes that $n \geq const L^2 \log M$, for a positive *const* not depending upon L or M. In other words, the rate R is of order $1/L$, which goes to 0 for large L. Correspondingly, results from these works cannot be directly applied to our communication problem.

As mentioned earlier, the idea of adapting techniques in compressed sensing to solve the communication problem began with Tropp [39]. However, since he used a condition similar to the irrepresentable condition discussed above, his results do not demonstrate communication at positive rates.

We also remark that conditions such as (9) are required by algorithms such as Lasso and OMP for providing uniform guarantees on support recovery. However, there are algorithms which provided guarantees with much weaker conditions on X. Examples include the iterative forward-backward algorithm [46] and least squares minimization using concave penalties [45]. Even though these results, when translated to our setting, do imply communication at positive rates is possible, a demonstration that rates up to capacity can be achieved has been lacking.

The second approach, as discussed above, is to assign a distribution for the X matrix and analyze performance after averaging over this distribution. Wainwright [44] considers X matrices with rows i.i.d. Normal(0, Σ), where Σ satisfies certain conditions, and shows that recovery is possible with the Lasso with n that is $\Omega(L \log M)$. In particular his results hold for the i.i.d. Gaussian ensembles that we consider here. Analogous results for the OMP was shown by [19], [26]. Another result in the same spirit of average case analysis is done by Candès and Plan [14] for the Lasso, where the authors assign a prior distribution to β and study the performance after averaging over this distribution. The X matrix is assumed to

satisfy a weaker form of the incoherence condition that holds with high-probability for i.i.d Gaussian designs, with n again of the right order.

A caveat in these discussions is that the aim of much (though not all) of the work on sparse signal recovery, compressed sensing, and term selection in linear statistical models is distinct from the purpose of communication alone. In particular rather than the non-zero coefficients being fixed according to a particular power allocation, the aim is to allow a class of coefficients vectors, such as that described above, and still recover their support and estimate the coefficient values. The main distinction from us being that our coefficient vectors belong to a finite set, of M^L elements, whereas in the above literature the class of coefficients vectors is almost always infinite. This additional flexibility is one of the reasons why an exact characterization of achieved rate has not been done in these works.

Another point of distinction is that majority of these works focus on exact recovery of the support of the true of coefficient vector β. As mentioned before, as our non-zeroes are quite small (of the order of $1/\sqrt{L}$), one cannot get exponentially small error probabilities for exact support recovery. Correspondingly, it is essential to relax the stipulation of exact support recovery and allow for a certain small fraction of mistakes (both false alarms and failed detection). There have been works by Reeves and Gastpar [33]–[35] that give lower bounds on the sample size n for approximate sparsity recovery. These works also provide results on orders of magnitude of n for approximate recovery of sparse signals for certain algorithms. However, to the best of our knowledge, there is still a need in the sparse signal recovery literature to provide precise relationships between sample size, mistake rates and error probabilities for algorithms such as Lasso or OMP.

Section II describes our adaptive successive decoder in full detail. Section III describes the computational resource required for the algorithm. Section IV presents the tools for the theoretical analysis of the algorithm, while section V presents the theorem for reliability of the algorithm. Computational illustrations are included in section VI. Section VII proves results for the function g_L of figure 2, required for the demonstrating that one can indeed achieve rates up to capacity. The appendix collects some auxiliary matters.

II. The Decoder

The algorithm we analyze is a modification of the algorithm described in subsection I-B. The main reason for the modification is due to the difficulty in analyzing the statistics $\mathcal{Z}^{res}_{k,j}$, for $j \in J_k$ and for steps $k \geq 2$.

The distribution of the statistic $\mathcal{Z}_{1,j}$, used in the first step, is easy, as will be seen below. This is because of the fact that the random variables

$$\{X_j,\ j \in J\} \quad \text{and} \quad Y$$

are jointly multivariate normal. However, this fails to hold for the random variables,

$$\{X_j,\ j \in J_k\} \quad \text{and} \quad R_k$$

used in forming $\mathcal{Z}^{res}_{k,j}$.

It is not hard to see why this joint Gaussianity fails. Recall that R_k may be expressed as,

$$R_k = Y - \sum_{j \in \mathsf{Dec}_{k-1}} \sqrt{P_j} X_j.$$

Correspondingly, since the event Dec_{k-1} is not independent of the X_j's, the quantities R_k, for $k \geq 2$, are no longer normal random vectors. It is for this reason the we introduce the following two modifications.

A. First Modification: Using a Combined Statistic

We overcome the above difficulty in the following manner. Recall that each

$$R_k = Y - F_1 - \cdots - F_{k-1}, \tag{10}$$

is a sum of Y and $-F_1, \ldots, -F_{k-1}$. Let $G_1 = Y$ and denote G_k, for $k \geq 2$, as the part of $-F_{k-1}$ that is orthogonal to the previous G_k's. In other words, perform Gram-Schmidt orthogonalization on the vectors $Y, -F_1, \ldots, -F_{k-1}$, to get $G_{k'}$, with $k' = 1, \ldots, k$. Then, from (10),

$$\frac{R_k}{\|R_k\|} = \mathsf{weight}_1 \frac{G_1}{\|G_1\|} + \mathsf{weight}_2 \frac{G_2}{\|G_2\|} + \ldots + \mathsf{weight}_k \frac{G_k}{\|G_k\|},$$

for some weights, denoted by $\mathsf{weight}_{k'} = \mathsf{weight}_{k',k}$, for $k' = 1, \ldots, k$. More specifically,

$$\mathsf{weight}_{k'} = \frac{R_k^{\mathrm{T}} G_{k'}}{\|R_k\| \|G_{k'}\|},$$

and,

$$\mathsf{weight}_1^2 + \cdots + \mathsf{weight}_k^2 = 1.$$

Correspondingly, the statistic $\mathcal{Z}^{res}_{k,j} = X_j^{\mathrm{T}} R_k / \|R_k\|$, which we want to use for k-th step detection, may be expressed as,

$$\mathcal{Z}^{res}_{k,j} = \mathsf{weight}_1\, \mathcal{Z}_{1,j} + \mathsf{weight}_2\, \mathcal{Z}_{2,j} + \cdots + \mathsf{weight}_{k-1}\, \mathcal{Z}_{k,j},$$

where,

$$\mathcal{Z}_{k,j} = X_j^T G_k / \|G_k\|. \tag{11}$$

Instead of using the statistic $\mathcal{Z}^{res}_{k,j}$, for $k \geq 2$, we find it more convenient to use statistics of the form,

$$\mathcal{Z}^{comb}_{k,j} = \lambda_{1,k}\, \mathcal{Z}_{1,j} + \lambda_{2,k}\, \mathcal{Z}_{2,j} + \cdots + \lambda_{k,k}\, \mathcal{Z}_{k,j}, \tag{12}$$

where $\lambda_{k',k}$, for $k' = 1, \ldots, k$ are positive weights satisfying,

$$\sum_{k'=1}^{k} \lambda_{k',k}^2 = 1.$$

For convenience, unless there is some ambiguity, we suppress the dependence on k and denote $\lambda_{k',k}$ as simply $\lambda_{k'}$. Essentially, we choose λ_1 so that it is a deterministic proxy for weight_1 given above. Similarly, $\lambda_{k'}$ is a proxy for $\mathsf{weight}_{k'}$ for $k' \geq 2$. The important modification we make, of replacing the random weight_k's by proxy weights, enables us to give an explicit characterization of the distribution of the statistic $\mathcal{Z}^{comb}_{k,j}$, which we use as a proxy for $\mathcal{Z}^{res}_{k,j}$ for detection of additional terms in successive iterations.

We now describe the algorithm after incorporating the above modification. For the time-being assume that for each k we have a vector of deterministic weights,

$$(\lambda_{k',k} : k' = 1, \ldots, k),$$

satisfying $\sum_{k'=1}^{k} \lambda_{k'}^2 = 1$, where recall that for convenience we denote $\lambda_{k',k}$ as $\lambda_{k'}$. Recall $G_1 = Y$.

For step $k = 1$, do the following

- For $j \in J$, compute

$$\mathcal{Z}_{1,j} = X_j^{\mathrm{T}} G_1 / \|G_1\|.$$

  To provide consistency with the notation used below, we also denote $\mathcal{Z}_{1,j}$ as $\mathcal{Z}_{1,j}^{comb}$.
- Update

$$\mathsf{dec}_1 = \{j \in J : \mathcal{Z}_{1,j}^{comb} \geq \tau\}, \tag{13}$$

  which corresponds to the set of decoded terms for the first step. Also let $\mathsf{Dec}_1 = \mathsf{dec}_1$. Update

$$F_1 = \sum_{j \in \mathsf{dec}_1} \sqrt{P_j} X_j.$$

This completes the actions of the first step. Next, perform the following steps for $k \geq 2$, with the number of steps k to be at most a pre-define value m.

- Define G_k as the part of $-F_{k-1}$ orthogonal to $G_1, \ldots, G_{k-1}$.
- For $j \in J_k = J - \mathsf{Dec}_{k-1}$, calculate

$$\mathcal{Z}_{k,j} = \frac{X_j^{\mathrm{T}} G_k}{\|G_k\|} \tag{14}$$

- For $j \in J_k$, compute the combined statistic using the above $\mathcal{Z}_{k,j}$ and $\mathcal{Z}_{k',j}$, $0 \leq k' \leq k-1$, given by,

$$\mathcal{Z}_{k,j}^{comb} = \lambda_1 \mathcal{Z}_{1,j} + \lambda_2 \mathcal{Z}_{2,j} + \cdots + \lambda_k \mathcal{Z}_{k,j},$$

  where the weights $\lambda_{k'} = \lambda_{k,k'}$, which we specify later, are positive and have sum of squares equal to 1.
- Update

$$\mathsf{dec}_k = \{j \in J_k : \mathcal{Z}_{k,j}^{comb} \geq \tau\}, \tag{15}$$

  which corresponds to the set of decoded terms for the k th step. Also let $\mathsf{Dec}_k = \mathsf{Dec}_{k-1} \cup \mathsf{dec}_k$, which is the set of terms detected after k steps.
- This completes the k th step. Stop if either L terms have been decoded, or if no terms are above threshold, or if $k = m$. Otherwise increase k by 1 and repeat.

As mentioned earlier, part of what makes the above algorithm work is our ability to assign deterministic weights $(\lambda_{k,k'} : k' = 1, \ldots, k)$, for each step $k = 1, \ldots, m$. To be able to do so, we need good control on the (weighted) sizes of the set of decoded terms Dec_k after step k, for each k. In particular, defining for each j, the quantity $\pi_j = P_j / P$, we define the size of the set Dec_k as size_k, where

$$\mathsf{size}_k = \sum_{j \in \mathsf{Dec}_k} \pi_j. \tag{16}$$

Notice that size_k is increasing in k, and is a random quantity which depends on the number of correct detections and false alarms in each step. As we shall see, we need to provide good upper and lower bounds for the $\mathsf{size}_1, \ldots, \mathsf{size}_{k-1}$ that are satisfied with high probability, to be able to provide deterministic weights of combination, $\lambda_{k',k}$, for $k' = 1, \ldots, k$, for the kth step.

It turns out that the existing algorithm does not provide the means to give good controls on the size_k's. To be able to do so, we need to further modify our algorithm.

B. The Second Modification: Pacing the Steps

As mentioned above, we need to get good controls on the quantity size_k, for each k, where size_k is defined as above. For this we modify the algorithm even further.

Assume that we have certain pre-specified values θ_k, for $k = 1, \ldots, m$. Here the θ_k's, which are between 0 and 1, are called the *pacing parameters*. Explicit expressions θ_k, which are taken to be strictly increasing in k, will be specified later on. The weights of combination,

$$(\lambda_{k',k} : k' = 1, \ldots, k),$$

for $k = 1, \ldots, m$, will be functions of these values.

For each k, denote

$$\mathsf{thresh}_k = \{j : \mathcal{Z}_{k,j}^{comb} \geq \tau\}.$$

For the algorithm described in the previous subsection, dec_k, the set of decoded terms for the k th step, was taken to be equal to thresh_k. We make the following modification:

For each k, instead of making dec_k to be equal to thresh_k, take dec_k to be a subset of thresh_k so that the total size of the of the decoded set after k steps, given by size_k, is near θ_k. The set dec_k is chosen by selecting terms in thresh_k, in decreasing order of their $\mathcal{Z}_{k,j}^{comb}$ values, until size_k nearly equals θ_k.

In particular, given size_{k-1}, one continues to add terms in dec_k, if possible, until

$$\theta_k - 1/L_\pi < \mathsf{size}_k \leq \theta_k. \tag{17}$$

Here $1/L_\pi = \max_\ell \pi_{(\ell)}$, is the maximum non-zero weights over all sections. It is a small term of order $1/L$ for the power allocations we consider.

Of course, the set of terms thresh_k might not be large enough to arrange for dec_k satisfying (17). Nevertheless, it is satisfied, provided

$$\mathsf{size}_{k-1} + \sum_{j \in \mathsf{thresh}_k} \pi_j \geq \theta_k,$$

or equivalently,

$$\sum_{j \in \mathsf{Dec}_{k-1}} \pi_j + \sum_{j \in J - \mathsf{Dec}_{k-1}} \pi_j 1_{\{\mathcal{Z}_{k,j}^{comb} \geq \tau\}} \geq \theta_k. \tag{18}$$

Here we use the fact that $J_k = J - \mathsf{Dec}_{k-1}$.

Our analysis demonstrates that we can arrange for an increasing sequence of θ_k, with θ_m near 1, such that condition (18) is satisfied for $k = 1, \ldots, m$, with high probability. Correspondingly, size_k is near θ_k for each k with high probability. In particular, size_m, the weighted size of the decoded set after the final step, is near θ_m, which is near 1.

We remark that in [8], an alternative technique for analyzing the distributions of $\mathcal{Z}_{k,j}^{comb}$, for $j \in J_k$, is pursued, which

does away with the above approach of pacing the steps. The technique in [8] provides uniform bounds on the performance for collection of random variables indexed by the vectors of weights of combination. However, since the pacing approach leads to cleaner analysis, we pursue it here.

III. Computational Resource

For the decoder described in section II, the vectors G_k can be computed efficiently using the Gram-Schmidt procedure. Further, as will be seen, the weights of combination are chosen so that, for each k,

$$\mathcal{Z}_{k,j}^{comb} = \sqrt{1-\lambda_{k,k}^2}\,\mathcal{Z}_{k-1,j}^{comb} + \lambda_{k,k}\mathcal{Z}_{k,j}.$$

This allows us to computed the statistic $\mathcal{Z}_{k,j}^{comb}$ easily from the previous combined statistic. Correspondingly, for simplicity we describe here the computational time of the algorithm in subsection I-B, in which one works with the residuals and accepts each term above threshold. Similar results hold for the decoder in section II.

The inner products requires order nLM multiply and adds each step, yielding a total computation of order $nLMm$ for m steps. As we shall see, the ideal number of steps m according to our bounds is of order $\log M$.

When there is a stream of strings Y arriving in succession at the decoder, it is natural to organize the computations in a parallel and pipelined fashion as follows. One allocates m signal processing chips, each configured nearly identically, to do the inner products. One such chip does the inner products with Y, a second chip does the inner products with the residuals from the preceding received string, and so on, up to chip m which is working on the final decoding step from the string received several steps before. After an initial delay of m received strings, all m chips are working simultaneously.

If each of the signal processing chips keeps a local copy of the dictionary X, alleviating the challenge of numerous simultaneous memory calls, the total computational space (memory positions) involved in the decoder is $nLMm$, along with space for LMm multiplier-accumulators, to achieve constant order computation time per received symbol. Naturally, there is the alternative of increased computation time with less space; indeed, decoding by serial computation would have runtime of order $nLMm$. Substituting $L = nR/\log M$ and m of order $\log M$, we may reexpress $nLMm$ as n^2M . This is the total computational resource required (either space or time) for the sparse superposition decoder.

In the computational space complexity of n^2M, it is important to note the dependence on the section size M. In our present analysis, this M would be have to be made undesirably large to achieve reliability if the rate R is pushed too close to capacity. Put affirmatively, it is preferable to choose M to be of a low order power of the codelength n, and note that the code will be reliable when the rate is kept at least order $1/\sqrt{\log M}$ below the capacity. It is the constant order computation time per received symbol, while maintaining a manageable amount of computational space (of order n^2M), that is the reason for us calling our decoder 'fast'.

IV. Analysis

Recall that we need to give controls on the random quantity $\hat{\delta}_{mis}$ given by (3). Our analysis leads to controls on the following weighted measures of correct detections and false alarms for a step. Recall that $\pi_j = P_j/P$, where recall that $P_j = P_{(\ell)}$ for any j in section ℓ. The π_j sums to 1 across j in sent, and sums to $M-1$ across j in other. Define in general

$$\hat{q}_k = \sum_{j \in \mathsf{sent} \cap \mathsf{dec}_k} \pi_j, \tag{19}$$

which provides a weighted measure for the number of correct detections in step k, and

$$\hat{f}_k = \sum_{j \in other \cap \mathsf{dec}_k} \pi_j \tag{20}$$

for the false alarms in step k. Bounds on $\hat{\delta}_{mis}$ can be obtained from the quantities $\hat{q}_k$ and $\hat{f}_k$ as we now describe.

As mentioned in Subsection I-B, after the algorithm is run for m steps there are two types of mistakes one could make in a section, namely an error and an erasure. Notice that,

$$\sum_{j \in section\ \ell} \left(1_{\{j \in \mathsf{sent} \cap \mathsf{Dec}_m^c\}} + 1_{\{j \in \mathsf{other} \cap \mathsf{Dec}_m\}}\right) \geq 2 1_{\{\text{error in section } \ell\}} + 1_{\{\text{erasure in section } \ell\}} \tag{21}$$

To see this, consider the three possible cases. In the case when the section ℓ has neither an error or an erasure, the right side of (21) would be zero. Next, when the section has an error, the right side of (21) would be 2. The left side would also be 2, with a contribution of 1 from the correct term (since it is in $\mathsf{sent} \cap \mathsf{Dec}_m^c$), and another 1 from the wrong term. Lastly, if the section has an erasure, the right side would be 1, while the left side of (21) would be at least 1. In the latter case, the left side of (21) would be greater than 1 if there are multiple terms in other that are selected.

Denote

$$\hat{\delta}_{wght} = \left(1 - \sum_{k=1}^{m} \hat{q}_k\right) + \sum_{k=1}^{m} \hat{f}_k. \tag{22}$$

An equivalent way of expressing $\hat{\delta}_{wght}$ is the sum of ℓ from 1 to L of,

$$\pi_{(\ell)} \sum_{j \in section\ \ell} \left(1_{\{j \in \mathsf{sent} \cap \mathsf{Dec}_m^c\}} + 1_{\{j \in \mathsf{other} \cap \mathsf{Dec}_m\}}\right),$$

where $\pi_j = \pi_{(\ell)}$ for j in section ℓ.

In the equal power allocation case, where $\pi_j = 1/L$, one has $\hat{\delta}_{mis} \leq \hat{\delta}_{wght}$. This can be seen from (21), and the expression of $\hat{\delta}_{mis}$ given by (3). For the power allocation (2) that we consider, bounds on $\hat{\delta}_{mis}$ are obtained by multiplying $\hat{\delta}_{wght}$ by the factor $\mathsf{snr}/(2\mathcal{C})$. To see this, notice that for a given weighted fraction, the maximum possible un-weighted fraction would be if we assume that all the failed detection or false alarms came from the section with the smallest weight. This would correspond to the section with weight $\pi_{(L)}$, where it is seen that $\pi_{(L)} = 2\mathcal{C}/(L\,\mathsf{snr})$. Accordingly, if δ_{wght} were

an upper bound on $\hat{\delta}_{wght}$ that is satisfied with high probability, we take

$$\delta_{mis} = \frac{\mathsf{snr}}{2\mathcal{C}}\,\delta_{wght}, \tag{23}$$

so that $\hat{\delta}_{mis} \le \delta_{mis}$ with high probability as well.

Next, we characterize, for $k \ge 1$, the distribution of $\mathcal{Z}_{k,j}$, for $j \in J_k$. As we mentioned earlier, the distribution of $\mathcal{Z}_{1,j}$ is easy to characterize. Accordingly, we do this separately in the next subsection. In subsection IV-B we provide the analysis for the distribution of $\mathcal{Z}^{comb}_{k,j}$, for $k \ge 2$.

A. Analysis of the First Step

In Lemma 3 below we derive the distributional properties of $(\mathcal{Z}_{1,j} : j \in J)$. Lemma 4, in the next subsection, characterizes the distribution of $(\mathcal{Z}_{k,j} : j \in J_k)$ for steps $k \ge 2$.

Define

$$C_{j,R} = n\,\pi_j\,\nu, \tag{24}$$

where recall that $\nu = P/(P+\sigma^2)$. For the constant power allocation case, π_j equals $1/L$. In this case $C_{j,R} = (R_0/R)\,2\log M$ is the same for all j.

For the variable power allocation (2), we have

$$\pi_j = e^{-2\mathcal{C}(\ell-1)/L}(1-e^{-2\mathcal{C}/L})/(1-e^{-2\mathcal{C}}),$$

for each j in section ℓ. Let

$$\tilde{\mathcal{C}} = (L/2)[1-e^{-2\mathcal{C}/L}], \tag{25}$$

which is essentially identical to $\mathcal{C}$ when L is large. Then for j in section ℓ, we have

$$C_{j,R} = (\tilde{\mathcal{C}}/R)\,e^{-2\mathcal{C}(\ell-1)/L}(2\log M). \tag{26}$$

Note, in this expression, the multiplier of $2\log M$ is at least 1 for $R \le \tilde{\mathcal{C}}$ and for small section indices ℓ. It dips below 1 as the section index grows.

We now are in a position to give the lemma for the distribution of $\mathcal{Z}_{1,j}$, for $j \in J$. The lemma below shows that each $\mathcal{Z}_{1,j}$ is distributed as a shifted normal, where the shift is approximately equal to $\sqrt{C_{j,R}}$ for any j in sent, and is zero for j in other. Accordingly, for a particular section, the maximum of the $\mathcal{Z}_{1,j}$, for $j \in$ other, is seen to be approximately $\sqrt{2\log M}$, since it is the maximum of $M-1$ independent standard normal random variables. Consequently, one would like $\sqrt{C_{j,R}}$ to be at least $\sqrt{2\log M}$ for the correct term in that section to be detected.

Lemma 3: For each $j \in J$, the statistic $\mathcal{Z}_{1,j}$ can be represented as

$$\sqrt{C_{j,R}}\,(\mathcal{X}_n/\sqrt{n})1_{\{j\ \mathsf{sent}\}} + N_{1,j},$$

where $N_1 = (N_{1,j} : j \in J_1)$ is multivariate normal Normal$(0, \Sigma_1)$, with $\Sigma_1 = I - \delta_1\delta_1^{\mathrm{T}}/P$, where $\delta_1 = \sqrt{\nu}\beta$.

Also,

$$\mathcal{X}_n^2 = \frac{\|Y\|^2}{\sigma_Y^2}$$

is a Chi-square n random variable that is independent of $N_1 = (N_{1,j} : j \in J)$. Here $\sigma_Y = \sqrt{P+\sigma^2}$ is the standard deviation of each coordinate of Y.

Proof: Recall that the X_j, for j in J, are independent Normal$(0, I)$ random vectors and that $Y = \sum_j \beta_j X_j + \varepsilon$, where the sum of squares of the β_j is equal to P.

The conditional distribution of each X_j given Y may be expressed as,

$$X_j = \beta_j\,Y/\sigma_Y^2 + U_j, \tag{27}$$

where U_j is a vector in $\mathbb{R}^N$ having a multivariate normal distribution. Denote $b = \beta/\sigma_Y$. It is seen that

$$U_j \sim N_n\left(0, (1-b_j^2)I\right),$$

where b_j is the j th coordinate of b.

Further, letting $U = [U_1 : \ldots : U_N]$, it follows from the fact that the rows of $[X : \epsilon/\sigma]$ are i.i.d, that the rows of the matrix U are i.i.d.

Further, for row i of U, the random variables $U_{i,j}$ and $U_{i,j'}$ have mean zero and expected product

$$1_{\{j=j'\}} - b_j b_{j'}.$$

In general, the covariance matrix of the ith row of U is given by Σ_1.

For any constant vector $\alpha \ne 0$, consider $U_j^T\alpha/\|\alpha\|$. Its joint normal distribution across terms j is the same for any such α. Specifically, it is a normal Normal$(0, \Sigma_1)$, with mean zero and the indicated covariances.

Likewise define $N_{1,j} = U_j^T Y/\|Y\|$. Conditional on Y, one has that jointly across j, these $N_{1,j}$ have the normal Normal$(0, \Sigma)$ distribution. Correspondingly, $N_1 = (N_{1,j} : j \in J)$ is independent of Y, and has a Normal$(0, \Sigma_1)$ distribution unconditionally.

Where this gets us is revealed via the representation of the inner product $\mathcal{Z}_{1,j} = X_j^T Y/\|Y\|$, which using (27), is given by,

$$\mathcal{Z}_{1,j} = \beta_j\,\frac{\|Y\|}{\sigma_Y^2} + N_{1,j}.$$

The proof is completed by noticing that for $j \in$ sent, one has $\sqrt{C_{j,R}} = \beta_j\sqrt{n}/\sigma_Y$. ∎

B. Analysis of Steps $k \ge 2$

We need the characterize the distribution of the statistic $\mathcal{Z}^{comb}_{k,j}$, $j \in J_k$, used in decoding additional terms for the kth step.

The statistic $\mathcal{Z}^{comb}_{k,j}$, $j \in J_k$, can be expressed more clearly in the following manner. For each $k \ge 1$, denote,

$$\mathcal{Z}_k = X^{\mathrm{T}}\frac{G_k}{\|G_k\|}.$$

Further, notice that $\mathcal{Z}^{comb}_{k,j}$ is simply the j th element of the vector

$$\mathcal{Z}^{comb}_k = \lambda_{k,1}\mathcal{Z}_1 + \lambda_{k,2}\mathcal{Z}_2 + \ldots, +\lambda_{k,k}\mathcal{Z}_k.$$

We remind that for step k we are only interested in elements $j \in J_k$, that is, those that were not decoded in previous steps.

Below we characterize the distribution of $\mathcal{Z}^{comb}_k$ conditioned on the what occurred on previous steps in the algorithm. More explicitly, we define $\mathcal{F}_{k-1}$ as

$$\mathcal{F}_{k-1} = (G_1,\, G_2, \ldots, G_{k-1}, \mathcal{Z}_1, \ldots, \mathcal{Z}_{k-1}), \tag{28}$$

or the associated σ-field of random variables. This represents the variables computed up to step $k-1$. Notice that from the knowledge of $\mathcal{Z}_{k'}$, for $k' = 1, ..., k-1$, one can compute $\mathcal{Z}_{k'}^{comb}$, for $k' < k$. Correspondingly, the set of decoded terms $\mathsf{dec}_{k'}$, till step $k-1$, is completely specified from knowledge of $\mathcal{F}_{k-1}$.

Next, note that in $\mathcal{Z}_k^{comb}$, only the vector $\mathcal{Z}_k$ does not belong to $\mathcal{F}_{k-1}$. Correspondingly, the conditional distribution of $\mathcal{Z}_k^{comb}$ given $\mathcal{F}_{k-1}$, is described completely by finding the distribution of $\mathcal{Z}_k$ given $\mathcal{F}_{k-1}$. Accordingly, we only need to characterize the conditional distribution of $\mathcal{Z}_k$ given $\mathcal{F}_{k-1}$.

Initializing with the distribution of $\mathcal{Z}_1$ derived in Lemma 3, we provide the conditional distributions

$$\mathcal{Z}_{k,J_k} = (\mathcal{Z}_{k,j} : j \in J_k),$$

for $k = 2, \ldots, n$. As in the first step, we show that the distribution of $\mathcal{Z}_{k,J_k}$ can be expressed as the sum of a mean vector and a multivariate normal noise vector $N_{k,J_k} = (N_{k,j} : j \in J_k)$. The algorithm will be arranged to stop long before n, so we will only need these up to some much smaller final $k = m$. Note that J_k is never empty because we decode at most L, so there must always be at least $(M-1)L$ remaining.

The following measure of correct detections in step, adjusted for false alarms, plays an important role in characterizing the distributions of the statistics involved in an iteration. Denote

$$\hat{q}_k^{adj} = \frac{\hat{q}_k}{1 + \hat{f}_k/\hat{q}_k}, \tag{29}$$

where $\hat{q}_k$ and $\hat{f}_k$ are given by (19) and (20).

In the lemma below we denote $\text{Normal}_{J_k}(0, \Sigma)$ to be multivariate normal distribution with dimension $|J_k|$, having mean zero and covariance matrix Σ, where Σ is an $|J_k| \times |J_k|$ dimensional matrix. Further, we denote β_{J_k} to be the subvector of β consisting of terms with indices in J_k.

Lemma 4: For each $k \geq 2$, the conditional distribution of $\mathcal{Z}_{k,j}$, for $j \in J_k$, given $\mathcal{F}_{k-1}$ has the representation

$$\sqrt{\hat{w}_k\, C_{j,R}}\, (\mathcal{X}_{d_k}/\sqrt{n})\, 1_{\{j \in \mathsf{sent}\}} + N_{k,j}. \tag{30}$$

Recall that $C_{j,R} = n\pi_j \nu$. Further, $\hat{w}_k = \hat{s}_k - \hat{s}_{k-1}$, which are increments of a series with total

$$\hat{w}_1 + \hat{w}_2 + \cdots + \hat{w}_k = \hat{s}_k = \frac{1}{1 - \hat{q}_{k-1}^{adj,tot}\, \nu},$$

where

$$\hat{q}_k^{adj,tot} = \hat{q}_1^{adj} + \cdots + \hat{q}_k^{adj}. \tag{31}$$

Here $\hat{w}_1 = \hat{s}_1 = 1$. The quantities $\hat{q}_k^{adj}$ is given by (29).

The conditional distribution $\mathbb{P}_{N_{k,J_k}|\mathcal{F}_{k-1}}$ is $\text{Normal}_{J_k}(0, \Sigma_k)$, where the covariance Σ_k has the representation

$$\Sigma_k = I - \delta_k \delta_k^T / P, \quad \text{where } \delta_k = \sqrt{\nu_k}\, \beta_{J_k}.$$

Here $\nu_k = \hat{s}_k \nu$.

Define $\sigma_k^2 = \hat{s}_{k-1}/\hat{s}_k$. The $\mathcal{X}_{d_k}$ term appearing in (30) is given by

$$\mathcal{X}_{d_k}^2 = \frac{\|G_k\|^2}{\sigma_k^2}.$$

Also, the distribution of $\mathcal{X}_{d_k}^2$ given $\mathcal{F}_{k-1}$, is chi-square with $d_k = n-k+1$ degrees of freedom, and further, it is independent of N_{k,J_k}.

The proof of the above lemma is considerably more involved. It is given in Appendix A. From the above lemma one gets that $\mathcal{Z}_{k,j}$ is the sum of two terms - the 'shift' term and the 'noise' term $N_{k,j}$. The lemma also provided that the noise term is normal with a certain covariance matrix Σ_k.

Notice that Lemma 4 applies to the case $k = 1$ as well, with $\mathcal{F}_0$ defined as empty, since $\hat{w}_k = \hat{s}_k = 1$. The definition of Σ_1 using the above lemma is the same as that given in Lemma 3. Also note that the conditional distribution of $(\mathcal{Z}_{k,j} : j \in J_k)$, as given in Lemma 4, depends on $\mathcal{F}_{k-1}$ only through the $\|G_1\|, \ldots, \|G_{k-1}\|$ and $(\mathcal{Z}_{k',j} : j \in J_{k'})$ for $k' < k$.

In the next subsection, we demonstrate that $N_{k,j}$, for $j \in J_k$, are very close to being independent and identically distributed (i.i.d.).

C. The Nearby Distribution

Recall that since the algorithm operates only on terms not detected previously, for the k step we are only interested in terms in J_k. The previous two lemmas specified conditional distributions of $\mathcal{Z}_{k,j}$, for $j \in J_k$. However, for analysis purposes we find it helpful to assign distributions to the $\mathcal{Z}_{k,j}$, for $j \in J - J_k$ as well. In particular, conditional on $\mathcal{F}_{k-1}$, write

$$\mathcal{Z}_{k,j} = \sqrt{\hat{w}_k\, C_{j,R}} \left(\frac{\mathcal{X}_{d_k}}{\sqrt{n}}\right) 1_{\{j \in \mathsf{sent}\}} + N_{k,j} \quad \text{for } j \in J.$$

Fill out of specification of the distribution assigned by $\mathbb{P}$, via a sequence of conditionals $\mathbb{P}_{N_k|\mathcal{F}_{k-1}}$ for $N_k = (N_{k,j} : j \in J)$, which is for all j in J, not just for j in J_k. For the variables N_{k,J_k} that we actually use, the conditional distribution is that of $\mathbb{P}_{N_{k,J_k}|\mathcal{F}_{k-1}}$ as specified in Lemmas 3 and 4. Whereas for the $N_{k,j}$ with $j \in J - J_k$, given $\mathcal{F}_{k-1}$, we conveniently arrange them to be independent standard normal. This definition is contrary to the true conditional distribution of $\mathcal{Z}_{k,j}$ for $j \in J - J_k$, given $\mathcal{F}_{k-1}$. However, it is a simple extension of the conditional distribution that shares the same marginalization to the true distribution of $(N_{k,j} : j \in J_k)$ given $\mathcal{F}_{k-1}$.

Further a simpler approximating distribution $\mathbb{Q}$ is defined. Define $\mathbb{Q}_{N_k|\mathcal{F}_{k-1}}$ to be independent standard normal. Also, like $\mathbb{P}$, the measure $\mathbb{Q}$ makes the $\mathcal{X}_{d_k}^2$ appearing in $\mathcal{Z}_{k,j}$, Chi-square$(n-k+1)$ random variables independent of N_k, conditional on $\mathcal{F}_{k-1}$.

In the following lemma we appeal to a sense of closeness of the distribution $\mathbb{P}$ to $\mathbb{Q}$, such that events exponentially unlikely under $\mathbb{Q}$ remain exponentially unlikely under the governing measure $\mathbb{P}$.

Lemma 5: For any event A that is determined by the random variables,

$$\|G_1\|, \ldots, \|G_k\| \text{ and } (\mathcal{Z}_{k',j} : j \in J_{k'}), \text{ for } k' \leq k, \tag{32}$$

one has

$$\mathbb{P}[A] \leq \mathbb{Q}[A] e^{kc_0},$$

where $c_0 = (1/2)\log(1 + P/\sigma^2)$.

For ease of exposition we give the proof in Appendix B. Notice that the set A is $\mathcal{F}_k$ measurable, since the random variables that A depends on are $\mathcal{F}_k$ measurable.

D. Separation Analysis

Our analysis demonstrates that we can give good lower bounds for $\hat{q}_k$, the weighted proportion of correct detection in each step, and good upper bounds on $\hat{f}_k$, which is the proportion of false alarms in each steps.

Denote the exception events

$$A_k = \{\hat{q}_k < q_k\} \quad \text{and} \quad B_k = \{\hat{f}_k > f_k\}.$$

Here the q_k and f_k are deterministic bounds for the proportion of correct detections and false alarms respectively, for each k. These will be specified in the subsequent subsection.

Assuming that we have got good controls on these quantities up to step $k-1$, we now describe our characterization of $\mathcal{Z}_{k,j}^{comb}$, for $j \in J_k$, used in detection for the kth step. Define the exception sets

$$A_{1,k-1} = \cup_{k'=1}^{k-1} A_{k'} \quad \text{and} \quad B_{1,k-1} = \cup_{k'=1}^{k-1} B_{k'}.$$

The manner in which the quantities $\hat{q}_1, \ldots, \hat{q}_k$ and $\hat{f}_1, \ldots \hat{f}_k$ arise in the distributional analysis of Lemma 4 is through the sum

$$\hat{q}_k^{adj,tot} = \hat{q}_1^{adj} + \cdots + \hat{q}_k^{adj}$$

of the adjusted values $\hat{q}_k^{adj} = \hat{q}_k/(1 + \hat{f}_k/\hat{q}_k)$. Outside of $A_{1,k-1} \cup B_{1,k-1}$, one has

$$\hat{q}_{k'}^{adj} \geq q_{k'}^{adj} \quad \text{for } k' = 1, \ldots, k-1, \tag{33}$$

where, for each k,

$$q_k^{adj} = q_k/(1 + f_k/q_k).$$

Recall that from Lemma 4 that,

$$\hat{w}_k = \frac{1}{1 - \hat{q}_{k-1}^{adj,tot}\nu} - \frac{1}{1 - \hat{q}_{k-2}^{adj,tot}\nu}.$$

From relation (33), one has $\hat{w}_{k'} \geq w_{k'}$, for $k' = 1, \ldots, k$, where $w_1 = 1$, and for $k > 1$,

$$w_k = \frac{1}{1 - q_{k-1}^{adj,tot}\nu} - \frac{1}{1 - q_{k-2}^{adj,tot}\nu}.$$

Here, for each k, we take

$$q_k^{adj,tot} = q_1^{adj} + \cdots + q_k^{adj}. \tag{34}$$

Using this w_k we define the corresponding vector of weights $(\lambda_{k',k} : k' = 1, \ldots, k)$, used in forming the statistics $\mathcal{Z}_{k,j}^{comb}$, as

$$\lambda_{k',k} = \sqrt{\frac{w_{k'}}{w_1 + w_2 + \cdots + w_k}}.$$

Given that the algorithm has run for $k-1$ steps, we now proceed to describe how we characterize the distribution of $\mathcal{Z}_{k,j}^{comb}$ for the kth step. Define the additional exception event

$$D_{1,k-1} = \cup_{k'=1}^{k-1} D_{k'}, \quad \text{with} \quad D_k = \{\mathcal{X}_{d_k}^2/n \leq 1-h\},$$

where $0 < h < 1$. Here the term $\mathcal{X}_{d_k}^2$ is as given in Lemma 4. It follows a Chi-square distribution with $d_k = n-k+1$ degrees of freedom. Define

$$E_{k-1} = A_{1,k-1} \cup B_{1,k-1} \cup D_{1,k-1}.$$

Notice that we have for $j \in \mathsf{sent}$ that

$$\mathcal{Z}_{k',j} = \sqrt{\hat{w}_{k'} C_{j,R}}\,(\mathcal{X}_{d_{k'}}/\sqrt{n}) + N_{k',j}$$

and for $j \in \mathsf{other}$, we have

$$\mathcal{Z}_{k',j} = N_{k',j},$$

for $k' = 1, \ldots, k$. Further, denote $C_{j,R,h} = C_{j,R}(1-h)$. Then on the set $E_{k-1}^c \cap D_k^c$, we have for $k' = 1, \ldots, k$ that

$$\mathcal{Z}_{k',j} \geq \sqrt{w_{k'}}\sqrt{C_{j,R,h}} \;+\; N_{k',j} \quad \text{for } j \in \mathsf{sent}.$$

Recall that,

$$\mathcal{Z}_{k,j}^{comb} = \lambda_1 \mathcal{Z}_{1,j} + \lambda_2 \mathcal{Z}_{2,j} + \cdots + \lambda_k \mathcal{Z}_{k,j},$$

where for convenience we denote $\lambda_{k',k}$ as simply $\lambda_{k'}$. Define for each k and $j \in J$, the combination of the noise terms by

$$N_{k,j}^{comb} = \lambda_1 N_{1,j} + \lambda_2 N_{2,j} + \cdots + \lambda_k N_{k,j}.$$

From the above one sees that, for $j \in \mathsf{other}$ the $\mathcal{Z}_{k,j}^{comb}$ equals $N_{k,j}^{comb}$, and for $j \in \mathsf{sent}$, on the set $E_{k-1}^c \cap D_k^c$, the statistic $\mathcal{Z}_{k,j}^{comb}$ exceeds

$$\left[\lambda_1\sqrt{w_1} + \cdots + \lambda_k\sqrt{w_k}\right]\sqrt{C_{j,R,h}} + N_{k,j}^{comb},$$

which is equal to

$$\sqrt{\frac{C_{j,R,h}}{1 - q_{k-1}^{adj,tot}\nu}} \;+\; N_{k,j}^{comb}.$$

Summarizing,

$$\mathcal{Z}_{k,j}^{comb} = N_{k,j}^{comb} \qquad \text{for } j \in \mathsf{other}$$

and, on the set $E_{k-1}^c \cap D_k^c$,

$$\mathcal{Z}_{k,j}^{comb} \geq \mathsf{shift}_{k,j} + N_{k,j}^{comb}, \qquad \text{for } j \in \mathsf{sent},$$

where

$$\mathsf{shift}_{k,j} = \sqrt{\frac{C_{j,R,h}}{1 - x_{k-1}\nu}},$$

with $x_0 = 0$ and $x_{k-1} = q_{k-1}^{adj,tot}$, for $k \geq 2$. Since the x_k's are increasing, the $\mathsf{shift}_{k,j}$'s increases with k. It is this increase in the mean shifts that helps in additional detections.

For each $j \in J$, set $H_{k,j}$ to be the event,

$$H_{k,j} = \left\{\mathsf{shift}_{k,j} 1_{\{j \in \mathsf{sent}\}} + N_{k,j} \geq \tau\right\}. \tag{35}$$

Notice that

$$H_{k,j} = \{\mathcal{Z}_{k,j}^{comb} \geq \tau\} \qquad \text{for } j \in \mathsf{other}. \tag{36}$$

On the set $E_{k-1}^c \cap D_k^c$, defined above, one has

$$H_{k,j} \subseteq \{\mathcal{Z}_{k,j}^{comb} \geq \tau\} \qquad \text{for } j \in \mathsf{sent}. \tag{37}$$

Using the above characterization of $\mathcal{Z}_{k,j}^{comb}$ we specify in the next subsection the values for θ_k, f_k and q_k. Recall that the quantity θ_k, which was defined is subsection II-B, gave controls on size_k, the size of the decoded set Dec_k after the k step.

E. Specification of f_k, θ_k, and q_k, for $k = 1, \ldots, m$

Recall from subsection IV-C that under the $\mathbb{Q}$ measure that $N_{k,j}$, for $j \in J$, are i.i.d. standard normal random variables. Define the random variable

$$\hat{f}_k^{up} = \sum_{j \in \mathsf{other}} \pi_j 1_{H_{k,j}}. \tag{38}$$

Notice that $\hat{f}_k \le \hat{f}_k^{up}$ since

$$\begin{aligned} \hat{f}_k &= \sum_{j \in \mathsf{dec}_k \cap \mathsf{other}} \pi_j \\ &\le \sum_{j \in \mathsf{other}} \pi_j 1_{\{\mathcal{Z}_{k,j}^{comb} \ge \tau\}}. \end{aligned} \tag{39}$$

The above inequality follows since dec_k is a subset of $\mathsf{thresh}_k = \{j : \mathcal{Z}_{k,j}^{comb} \ge \tau\}$ by construction. Further (39) is equal to $\hat{f}_k^{up}$ using (36).

The expectation of $\hat{f}_k^{up}$ under the $\mathbb{Q}$-measure is given by,

$$\mathbb{E}_{\mathbb{Q}}\left(\hat{f}_k^{up}\right) = (M-1)\bar{\Phi}(\tau),$$

where $\bar{\Phi}(\tau)$ is the upper tail probability of a standard normal at τ. Here we use the fact that the $H_{k,j}$, for $j \in \mathsf{other}$, are i.i.d Bernoulli $\bar{\Phi}(\tau)$ under the $\mathbb{Q}$-measure and that $\sum_{j \in \mathsf{other}} \pi_j$ is equal to $(M-1)$.

We assume that the threshold τ is of the form,

$$\tau = \sqrt{2 \log M} + a, \tag{40}$$

with a positive a specified in subsection VII-D. Define $f^* = (M-1)\bar{\Phi}(\tau)$, which is the expectation of $\hat{f}_k^{up}$ from above. One sees that

$$f^* \le \frac{\exp\left\{-a\sqrt{2\log M} - (1/2)a^2\right\}}{(\sqrt{2\log M} + a)\sqrt{2\pi}}, \tag{41}$$

using the form for the threshold τ in (40). We also use that $\bar{\Phi}(x) \le \phi(x)/x$ for positive x, with ϕ being the standard normal density. We take $f_k = f$ to be a value greater than f^*. We express it in the form

$$f = \rho f^*,$$

with a constant factor $\rho > 1$. This completes the specification of the f_k.

Next, we specify the θ_k used in pacing the steps. Denote the random variable,

$$\hat{\theta}_k = \sum_{j \text{ sent}} \pi_j 1_{H_{k,j}}. \tag{42}$$

Likewise, define θ_k^* as the expectation of $\hat{\theta}_k$ under the $\mathbb{Q}$ measure. Using (35), one has

$$\theta_k^* = \sum_{j \text{ sent}} \pi_j \bar{\Phi}(-\mu_{k,j}),$$

where $\mu_{k,j} = \mathsf{shift}_{k,j} - \tau$. Like before, we take θ_k to be a value less than θ_k^*. More specifically, we take

$$\theta_k = \theta_k^* - \eta \tag{43}$$

for a positive η.

This specification of θ_k^*, and the related θ_k, is a recursive definition in the following way: Notice that θ_k^* equals the function $g_L(x)$, given by (4), evaluated at $x_{k-1} = q_{k-1}^{adj,tot}$, with $x_0 = 0$. We define the target detection rate for the k th step, given by q_k, as

$$q_k = \theta_k - \theta_{k-1} - 1/L_\pi - f, \tag{44}$$

with θ_0 taken to zero. Here, $1/L_\pi = \max_{\ell=1,\ldots,L} \pi_{(\ell)}$ is a quantity of order $1/L$. Thus the q_k are specified from the θ_k and f. Using the expression of $x_{k-1} = q_{k-1}^{adj,tot}$ in (34), along with (44), one gets that θ_k^* is a function of $\theta_1, \ldots, \theta_{k-1}$ and f.

For instance, in the constant power allocation case $C_{j,R,h} = (R_0(1-h)/R)\, 2\log M$, is the same for all j. This makes $\mathsf{shift}_{k,j}$ the same for each j. Consequently, one has $\theta_k^* = \bar{\Phi}(-\mu(x_{k-1}))$, where $\mu(x) = \sqrt{1/(1-xv)}\sqrt{C_{j,R,h}} - \tau$. It obeys the recursion $\theta_k^* = g_L(x)$ evaluated at $x_{k-1} = q_{k-1}^{adj,tot}$, with $g_L(x) = \bar{\Phi}(-\mu(x))$.

F. Building Up the Total Detection Rate

The previous section demonstrated the importance of the function $g_L(x)$, given by (4). This function is defined on $[0, 1]$ and take values in the interval $(0, 1)$. Recall from subsection II-B, on pacing the steps, that the quantities θ_k are closely related to the proportion of sections correctly detected after k steps, if we ignore false alarm effects. Consequently, to ensure sufficient correct detections one would like the θ_k to increase with k to a value near 1. Through the recursive definition of θ_k, this amounts to ensuring that the function $g_L(x)$ is greater than x for an interval $[0, x_r]$, with x_r preferably near 1.

Definition: A function $g(x)$ is said to be *accumulative* for $0 \le x \le x_r$ with a positive gap, if

$$g(x) - x \ge \mathsf{gap}$$

for all $0 \le x \le x_r$. Moreover, the decoder is *accumulative* with a given rate and power allocation if corresponding function $g_L(x)$ satisfies this property for given x_r and positive gap.

To detail the progression of the θ_k consider the following lemma.

Lemma 6: Assume $g(x)$ is accumulative on $[0, x_r]$ with a positive gap, and η is chosen so that $\mathsf{gap} - \eta$ is positive. Further, assume

$$f \le (\mathsf{gap} - \eta)^2/8 - 1/(2L_\pi). \tag{45}$$

Then, one can arrange for an m so that the θ_k, for $k = 1, \ldots, m$, defined by (43), are increasing and

$$\theta_m \ge x_r + \mathsf{gap} - \eta.$$

Moreover, the number of steps m is at most $2/(\mathsf{gap} - \eta)$.

The proof of Lemma 6 is given in Appendix C. We now proceed to describe how we demonstrate the reliability of the algorithm using the quantities chosen above.

V. Reliability of the Decoder

We are interested in demonstrating that the probability of the event $A_{1,m} \cup B_{1,m}$ is small. This ensures that for each step k, where k ranges from 1 to m, the proportion of correct

detections $\hat{q}_k$ is at least q_k, and the proportion of false alarms $\hat{f}_k$ is at most $f_k = f$. We do this by demonstrating that the probability of the set

$$E_m = A_{1,m} \cup B_{1,m} \cup D_{1,m}$$

is exponentially small. The following lemma will be useful in this regard. Recall that

$$A_{1,m} = \cup_{k=1}^{m} \{\hat{q}_k < q_k\} \quad \text{and} \quad B_{1,m} = \cup_{k=1}^{m} \{\hat{f}_k > f\}.$$

Lemma 7: Let θ_k, q_k and f be as defined in subsection IV-E. Denote

$$\tilde{A}_{1,m} = \cup_{k=1}^{m} \{\hat{\theta}_k < \theta_k\} \quad \text{and} \quad \tilde{B}_{1,m} = \cup_{k=1}^{m} \{\hat{f}_k^{up} > f\}.$$

Then,

$$E_m \subseteq \tilde{A}_{1,m} \cup \tilde{B}_{1,m} \cup D_{1,m}.$$

For ease of exposition we provide the proof of this lemma in Appendix D. The lemma above described how we control the probability of the exception set E_m.

We demonstrate that the probability of E_m is exponentially small by showing that the probability of $\tilde{A}_{1,m} \cup \tilde{B}_{1,m} \cup D_{1,m}$, which contains E_m, is exponentially small. Also, notice that outside the set E_m, the weighted fraction of failed detection and false alarms, denoted by $\hat{\delta}_{wght}$ in (22), is bounded by

$$\left(1 - \sum_{k=1}^{m} q_k\right) + mf,$$

which, after recalling the definition of q_k in (44), can also be expressed as,

$$1 - \theta_m + 2mf + m/L_\pi. \tag{46}$$

Now, assume that g_L is accumulative on $[0, x_r]$ with a positive gap. Then, from Lemma 6, for $\eta <$ gap, and $f > f^*$ satisfying (45), one has that (46) is upper bounded by

$$\delta_{wght} = (1 - x_r) - (\text{gap} - \eta)/2, \tag{47}$$

using the bounds on f, θ_m and m given in the lemma. Consequently, $\hat{\delta}_{mis}$ the mistake rate after m steps, given by (3), is bounded by δ_{mis} outside of $\tilde{A}_{1,m} \cup \tilde{B}_{1,m} \cup D_{1,m}$, where,

$$\delta_{mis} = \frac{\text{snr}}{2\mathcal{C}}[(1 - x_r) - (\text{gap} - \eta)/2], \tag{48}$$

via (23). We then have the following theorem regarding the reliability of the algorithm.

Theorem 8: Let the conditions of Lemma 6 hold, and let δ_{mis} be as in (48). Then,

$$\mathbb{P}(\hat{\delta}_{mis} > \delta_{mis}) \leq me^{-2L_\pi \eta^2 + mc_0} + me^{-L_\pi f D(\rho)/\rho} + me^{-(n-m+1)h^2/2}.$$

Here the quantities η and ρ are as defined in subsection IV-E, and c_0 is as given in Lemma 5. Also $D(\rho) = \rho \log \rho - (\rho - 1)$.

Proof of Theorem 8: From Lemma 7, and the arguments above, the event $\{\hat{\delta}_{mis} > \delta_{mis}\}$ is contained in the event

$$\tilde{A}_{1,m} \cup \tilde{B}_{1,m} \cup D_{1,m}.$$

Consequently, we need to control the probability of the above three events under the $\mathbb{P}$ measure.

We first control the probability of the event $D_{1,m}$, which is the union of Chi-square events $D_k = \{\mathcal{X}_{d_k}^2/n < 1 - h\}$. Now the event D_k can be expressed as $\{\mathcal{X}_{d_k}^2/d_k < 1 - h_k\}$, where $h_k = (nh - k + 1)/(n - k + 1)$. Using a standard Chernoff bound argument, one gets that

$$\mathbb{P}(D_k) \leq e^{-(n-k+1)h_k^2/2}.$$

The exponent in the above is at least $(n - k + 1)h^2/2 - kh$. Consequently, as $k \leq m$, one gets, using a union bound that

$$\mathbb{P}(D_{1,m}) \leq me^{-(n-m+1)h^2/2 + mh}.$$

Next, lets focus on the event $\tilde{B}_{1,m}$, which is the union of events $\{\hat{f}_k^{up} > f\}$. Divide $\hat{f}_k^{up}$, f, by $M-1$ to get $\hat{p}_k$, p respectively. Consequently, $\tilde{B}_{1,m}$ is also the union of the events $\{\hat{p}_k > p\}$, for $k = 1, \ldots, m$, where

$$\hat{p}_k = \frac{1}{M-1} \sum_{j \in other} \pi_j \, 1_{H_{k,j}},$$

and $p = f/(M-1)$, with $f = \rho f^*$.

Recall, as previously discussed, for j in other, the event $H_{k,j}$ are i.i.d. Bernoulli(p^*) under the measure $\mathbb{P}$, where $p^* = f^*/(M-1)$. Consequently, from by Lemma 13 in the Appendix E, the probability of the event $\{\hat{p}_k \geq p\}$ is less than $e^{-L_\pi (M-1) D(p \| p^*)}$. Therefore,

$$\mathbb{P}(\tilde{B}_{1,m}) \leq me^{-L_\pi (M-1) D(p \| p^*)}.$$

To handle the exponents $(M-1)D(p\|p^*)$ at the small values p and p^*, we use the Poisson lower bound on the Bernoulli relative entropy, as shown in Appendix F. This produces the lower bound $(M-1)[p \log p/p^* + p^* - p]$, which is equal to

$$f \log f/f^* + f^* - f.$$

We may write this as $f^* D(\rho)$, or equivalently $f D(\rho)/\rho$, where the functions $D(\rho)$ and $D(\rho)/\rho = \log \rho + 1 - 1/\rho$ are increasing in ρ.

Lastly, we control the probability of the event $\tilde{A}_{1,m}$, which the is union of the events $\{\hat{\theta}_k < \theta_k\}$, where

$$\hat{\theta}_k = \sum_{j \in \text{sent}} \pi_j H_{k,j}.$$

We first bound the probability under the $\mathbb{Q}$ measure. Recall that under $\mathbb{Q}$, the $H_{k,j}$, for $j \in$ sent, are independent Bernoulli, with the expectation of $\hat{\theta}_k$ being θ_k^*. Consequently, using Lemma 13 in Appendix E, we have

$$\mathbb{Q}(\hat{\theta}_k < \theta_k) \leq e^{-L_\pi D(\theta_k \| \theta_k^*)}.$$

Further, by the Pinsker-Csiszar-Kulback-Kemperman inequality, specialized to Bernoulli distributions, the expressions $D(\theta_k \| \theta_k^*)$ in the above exceeds $2(\theta_k - \theta_k^*)^2$, which is $2\eta^2$, since $\theta_k^* - \theta_k = \eta$.

Correspondingly, one has

$$\mathbb{Q}(\tilde{A}_{1,m}) \leq me^{-L_\pi 2\eta^2}.$$

Now, use the fact that the event $\tilde{A}_{1,m}$ is $\mathcal{F}_m$ measurable, along with Lemma 5, to get that,

$$\mathbb{P}(\tilde{A}_{1,m}) \leq me^{-L_\pi 2\eta^2 + m c_0}.$$

This completes the proof of the lemma.

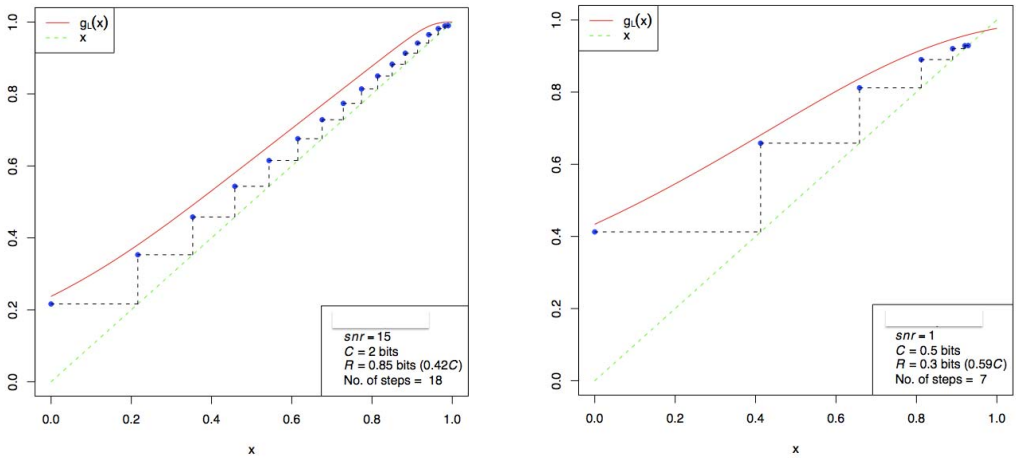

Fig. 3. Plots demonstrating progression of our algorithm. (Plot on left) $snr = 15$. The weighted (unweighted) detection rate is 0.995 (0.985) for a failed detection rate of 0.014 and the false alarm rate is 0.005. (Plot on right) $snr = 1$. The detection rate (both weighted and un-weighted) is 0.944 and the false alarm and failed detection rates are 0.016 and 0.055 respectively. Here $L = M = 2^{16}$.

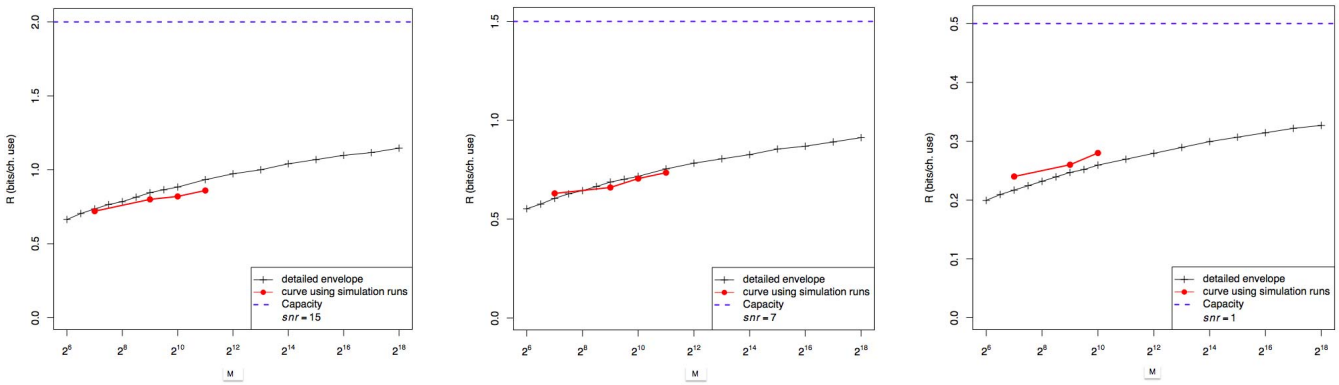

Fig. 4. Plots of achievable rates as a function of M for snr values of 15, 7 and 1. Section error rate is controlled to be between 9 and 10%. For the curve using simulation runs the error probability of making more than 10% section mistakes is taken to be 10^{-3}.

VI. Computational Illustrations

We illustrate in two ways the performance of our algorithm. First, for fixed values L, M, snr and rates below capacity we evaluate detection rate as well as probability of exception set p_e using the theoretical bounds given in Theorem 8. Plots demonstrating the progression of our algorithm are also shown. These highlight the crucial role of the function g_L in achieving high reliability.

Figure 3 presents the results of computation using the reliability bounds of Theorem 8 for fixed L and M and various choices of snr and rates below capacity. The dots in these figures denotes θ_k, for each k.

For illustrative purposes we take $M = 2^{16}$, $L = M$ and snr values of 1, 7 and 15, or, 0, 8.5, 11.8 dB respectively. The probability of error p_e is set to be near 10^{-3}. For each snr value the maximum rate, over a grid of values, for which the error probability is less than p_e is determined. With snr $= 1$ (Fig. 3), this rate R is 0.3 bits which is 59% of capacity. When snr is 7 (Fig. 2) and 15 (Fig. 3), these rates correspond to 49% and 42% of their corresponding capacities.

For the above computations we chose power allocations of the form

$$P_{(\ell)} \propto \max\{e^{-2\gamma\, l/L}, \text{cut}\},$$

with $0 \le \gamma \le C$, and $u > 0$. Here the choices of a, u and γ are made, by computational search, to minimize the resulting sum of false alarms and failed detections, as per our bounds. In the snr $= 1$ case, the optimum γ is 0, so we have constant power allocation in this case. In the other two cases, there is variable power across most of the sections. The role of a positive u being to increase the relative power allocation for sections with low weights.

Figure 4 gives plots of achievable rates as a function of M. For each M, the points on the detailed envelope correspond to the numerically evaluated maximum inner code rate for which the section error is between 9 and 10%. Here we assume L to be large, so that the θ_k and f_k are replaced by the expected values θ_k^* and f^*, respectively. We also take $h = 0$. This gives an idea about the best possible rates for a given snr and section error rate.

For the simulation curve, L was fixed at 100 and for given snr, M, and rate values, 10^4 runs of our algorithm

were performed. The maximum rate over the grid of values satisfying section error rate of less than 10% except in 10 replicates, (corresponding to an estimated p_e of 10^{-3}) is shown in the plots. Interestingly, even for such small values of L, the curve is is quite close to the detailed envelope curve, showing that our theoretical bounds are quite conservative.

VII. Achievable Rates Approaching Capacity

We demonstrate analytically that rates R moderately close to $\mathcal{C}$ are attainable by showing that the function $g_L(x)$ providing the updates for the fraction of correctly detected terms is indeed accumulative for suitable x_r and gap. Then the reliability of the decoder can be established via Theorem 8. In particular, the matter of normalization of the weights $\pi_{(\ell)}$ is developed in subsection VII-A. An integral approximation $g(x)$ to the sum $g_L(x)$ is provided in subsection VII-B, and in subsection VII-C we show that it is accumulative. Subsection VII-D addresses the issue of control of parameters that arise in specifying the code. In subsection VII-E, we give the proof of Theorem 2.

A. Variable Power Allocations

As mentioned earlier, we consider power allocations $P_{(\ell)}$ proportional to $e^{-2\mathcal{C}\ell/L}$. The function $g_L(x)$, given by (4), may also be expressed as

$$g_L(x) = \sum_{\ell=1}^{L} \pi_{(\ell)}\, \Phi(\mu(x, u_\ell\, \mathcal{C}'/R)),$$

where $\pi_{(\ell)} = P_{(\ell)}/P$, and,

$$\mu(x,u) = (\sqrt{u/(1-x\nu)} - 1)\sqrt{2\log M} - a$$

and

$$u_\ell = e^{-2\mathcal{C}(\ell-1)/L} \quad \text{and} \quad \mathcal{C}' = \tilde{\mathcal{C}}(1-h),$$

with $\tilde{\mathcal{C}}$ as in (25). Here we use the fact that $C_{j,R}$, for the above power allocation, is given by $u_\ell\, \tilde{\mathcal{C}}/R$ if j is in section ℓ, as demonstrated in (26).

Further, notice that $\pi_{(\ell)} = u_\ell/sum$, with $sum = \sum_{\ell=1}^{L} u_\ell$. One sees that $sum = L\nu/(2\tilde{\mathcal{C}})$, with $\nu = P/(P+\sigma^2)$. Using this one gets that

$$g_L(x) = \frac{2\mathcal{C}}{\nu L} \sum_{\ell=1}^{L} u_\ell\, \Phi\big(\mu(x, u_\ell \mathcal{C}'/R)\big). \tag{49}$$

B. Formulation and Evaluation of the Integral $g(x)$

Recognize that the sum in (49) corresponds closely to an integral. In each interval $\frac{\ell-1}{L} \le t < \frac{\ell}{L}$ for ℓ from 1 to L, we have $e^{-2\mathcal{C}\frac{\ell-1}{L}}$ at least $e^{-2\mathcal{C}t}$. Consequently, $g_L(x)$ is greater than $g(x)$ where

$$g(x) = \frac{2\mathcal{C}}{\nu} \int_0^1 e^{-2\mathcal{C}t} \Phi\big(\mu(x, e^{-2\mathcal{C}t}\mathcal{C}'/R)\big) dt. \tag{50}$$

The $g_L(x)$ and $g(x)$ are increasing functions of x on $[0,1]$.

Let's provide further characterization and evaluation of the integral $g(x)$. Let

$$z_x^{low} = \mu(x, (1-\nu)\,\mathcal{C}'/R) \quad \text{and} \quad z_x^{max} = \mu(x, \mathcal{C}'/R).$$

Further, let $\delta_a = a/\sqrt{2\log M}$. For emphasis we write out that $z_x = z_x^{low}$ takes the form

$$z_x = \left[\frac{\sqrt{(1-\nu)\mathcal{C}'/R}}{\sqrt{1-x\nu}} - (1+\delta_a) \right] \sqrt{2\log M}. \tag{51}$$

Change the variable of integration in (50) from t to $u = e^{-2\mathcal{C}t}$. Observing that $e^{-2\mathcal{C}} = 1-\nu$, one sees that

$$g(x) = \frac{1}{\nu} \int_{1-\nu}^{1} \Phi\big(\mu(x, u\,\mathcal{C}'/R)\big) du.$$

Now since

$$\Phi(\mu) = \int 1_{\{z \le \mu\}}\, \phi(z)\, dz,$$

it follows that

$$g(x) = \int\int 1_{\{u_{cut} \le u \le 1\}} 1_{\{z \le \mu(x, u\,\mathcal{C}'/R)\}}\, \phi(z)\, dz\, du/\nu. \tag{52}$$

In (52), the inequality

$$z \le \mu(x, u\,\mathcal{C}'/R)$$

is the same as

$$\sqrt{u} \;\ge\; \sqrt{u_x R/\mathcal{C}'}\,\big(1 + (z+a)/\sqrt{2\log M}\,\big),$$

provided $z_x^{low} \le z \le z_x^{max}$. Here $u_x = 1 - x\nu$. Thereby, for all z, the length of this interval of values of u can be written as

$$\left[1 - \max\left\{ u_x \frac{R}{\mathcal{C}'} \Big(1 + \frac{z+a}{\sqrt{2\log M}}\Big)_+^2, 1-\nu \right\} \right]_+ .$$

Thus $g(x)$ is equal to,

$$\frac{1}{\nu} \int \left[1 - \max\left\{ u_x \frac{R}{\mathcal{C}'} \Big(1 + \frac{z+a}{\sqrt{2\log M}}\Big)_+^2, 1-\nu \right\} \right]_+ \phi(z) dz. \tag{53}$$

Lemma 9: *Derivative evaluation.* The derivative $g'(x)$ may be expressed as

$$\frac{R}{\mathcal{C}'} \int_{z_x^{low}}^{z_x^{max}} \big(1 + \delta_a + \delta_z\big)^2 \phi(z) dz. \tag{54}$$

Further, if

$$R = \frac{\mathcal{C}'}{[(1+\delta_a)^2(1 + r/\log M)]}, \tag{55}$$

with $r \ge r_0$, where

$$r_0 = \frac{1}{2(1+\delta_a)^2}, \tag{56}$$

then the difference $g(x) - x$ is a decreasing function of x.

Proof: The integrand in (53) is continuous and piecewise differentiable in x, and its derivative is the integrand in (54).

Further, (54) is less than,

$$\frac{R}{\mathcal{C}'} \int_{-\infty}^{\infty} \big(1 + \delta_a + \delta_z\big)^2 \phi(z) dz = \frac{R}{\mathcal{C}'} \Big[(1+\delta_a)^2 + 1/(2\log M) \Big],$$

which is less than 1 for $r \geq r_0$. Consequently, $g(x) - x$ is decreasing as it has a negative derivative. ∎

Corollary 10: *A lower bound.* The function $g(x)$ is at least

$$g_{low}(x) = \frac{1}{\nu}\int_{z_x^{low}}^{\infty}\Big[1-(R/\mathcal{C}')u_x\big(1+(z+a)/\sqrt{2\log M}\big)^2\Big]\phi(z)dz.$$

This $g_{low}(x)$ is equal to

$$\Phi(z_x) + \Big[x + \delta_R\frac{u_x}{\nu}\Big]\big[1-\Phi(z_x)\big] - 2(1+\delta_a)\frac{R}{\mathcal{C}'}\frac{u_x}{\nu}\frac{\phi(z_x)}{\sqrt{2\log M}} - \frac{R}{\mathcal{C}'}\frac{u_x}{\nu}\frac{z_x\phi(z_x)}{2\log M}. \quad (57)$$

where

$$\delta_R = \frac{r - r_0}{\log M + r}.$$

Moreover, this $g_{low}(x)$ has derivative $g'_{low}(x)$ given by

$$\frac{R}{\mathcal{C}'}\int_{z_x}^{\infty}\big(1+\delta_a+\delta_z\big)^2\phi(z)dz.$$

Proof: The integral expressions for $g_{low}(x)$ are the same as for $g(x)$ except that the upper end point of the integration extends beyond z_x^{max}, where the integrand is negative. The lower bound conclusion follows from this negativity of the integrand above z_x^{max}. The evaluation of $g_{low}(x)$ is fairly straightforward after using $z\phi(z) = -\phi'(z)$ and $z^2\phi(z) = \phi(z) - (z\phi(z))'$. Also use that $\Phi(z)$ tends to 1, while $\phi(z)$ and $z\phi(z)$ tend to 0 as $z \to \infty$. This completes the proof of Corollary 10. ∎

Remark: What we gain with this lower bound is simplification because the result depends on x only through $z_x = z_x^{low}$.

C. Showing $g(x)$ is Greater Than x

The preceding subsection established that $g_L(x) - x$ is at least $g_{low}(x) - x$. We now show that

$$h_{low}(x) = g_{low}(x) - x,$$

is at least a positive value, which we denote as gap, on an interval $[0, x_r]$, with x_r suitably chosen.

Recall that $z_x = z_x^{low}$, given by (51), is a strictly increasing function of x, with values in the interval $I_0 = [z_0, z_1]$ for $0 \leq x \leq 1$. For values z in I_0, let $x = x(z)$ be the choice for which $z_x = z$. With the rate R of the form (55), let x_r be the value of x for which z_x is 0. One finds that x_r satisfies,

$$1 - x_r = \frac{1}{\mathsf{snr}}\left[\frac{r}{\log M}\right]. \quad (58)$$

We now show that $h_{low}(x)$ is positive on $[0, x_r]$, for r at least a certain value, which we call r_1.

Lemma 11: *Positivity of $h_{low}(x)$ on $[0, x_r]$.* Let rate R be of the form (55), with $r > r_1$, where

$$r_1 = r_0/2 + \frac{\sqrt{\log M}}{\sqrt{\pi}(1+\delta_a)}. \quad (59)$$

Then, for $0 \leq x \leq x_r$ the difference $h_{low}(x)$ is greater than or equal to

$$\mathsf{gap} = \frac{1}{\mathsf{snr}}\left[\frac{r - r_1}{\log M}\right]. \quad (60)$$

Proof of Lemma 11: The function $g(x)$ has lower bound $g_{low}(x)$. By Corollary 10, $g_{low}(x)$ has derivative bounded by

$$\int_{-\infty}^{\infty}\frac{\big(1+\delta_a+\delta_z\big)^2}{(1+\delta_a)^2(1+r/\log M)}\phi(z)dz = \frac{(1+r_0/\log M)}{(1+r/\log M)},$$

which is less than 1 for $r \geq r_0$. Thus $g_{low}(x) - x$ is decreasing as it has a negative derivative.

To complete the proof, evaluate $g_{low}(x) - x$ at the point $x = x_r$. The point x_r is the choice where $z_x = 0$. After using (57), it is seen that the value $g_{low}(x_r) - x_r$ is equal to gap, where gap is given by (60).

D. Choices of a and r That Control the Overall Rate Drop

Here we focus on the evaluation of a and r that optimize our summary expressions for the rate drop, based on the lower bounds on $g_L(x) - x$. Recall that the rate of our inner code is

$$R = \mathcal{C}\frac{1-h}{(1+\delta_a)^2(1+r/\log M)}.$$

Now, for $r > r_1$, the function $g_L(x)$ is accumulative on $[0, x_r]$, with positive gap given by (60). Notice that r_1, given by (59), satisfies,

$$r_1 \leq 1/4 + \frac{\sqrt{\log M}}{\sqrt{\pi}}. \quad (61)$$

Consequently, from Theorem 8, with high reliability, the total fraction of mistakes $\hat{\delta}_{mis}$ is bounded by

$$\delta_{mis} = \frac{\mathsf{snr}}{2\mathcal{C}}\left[(1 - x_r) - (\mathsf{gap} - \eta)/2\right].$$

If the outer Reed-Solomon code has distance designed to be at least δ_{mis} then any occurrences of a fraction of mistakes less than δ_{mis} are corrected. The overall rate of the code is R_{total}, which is at least $(1 - \delta_{mis})R$.

Sensible values of the parameters a and r can be obtained by optimizing the above overall rate under a presumption of small error probability, using simplifying approximations of our expressions. Reference values (corresponding to large L) are obtained by considering what the parameters become with $\eta = 0$, $f = f^*$, and $h = 0$.

Notice that a is related to f^* via the bound (41). Set a so that

$$a\sqrt{2\log M} = \log\Big[1/\big(f^*\sqrt{2\pi}\sqrt{2\log M}\big)\Big]. \quad (62)$$

We take f^* as $\mathsf{gap}^2/8$ as per Lemma 6. Consequently a will depend on r via the expression of gap given by (60).

Next, using the expressions for $1 - x_r$ and gap, along with $\eta = 0$, yields a simplified approximate expression for the mistake rate given by

$$\delta_{mis} = \frac{r + r_1}{4\mathcal{C}\log M}.$$

Accordingly, the overall communication rate may be expressed as,

$$R_{total} = \left(1 - \frac{r + r_1}{4\mathcal{C}\log M}\right)\frac{\mathcal{C}}{(1+\delta_a)^2(1+r/\log M)}.$$

As per these calculations (see [25] for details) we find it appropriate to take r to be r^*, where

$$r^* = r_1 + 2/(1 + 1/\mathcal{C}).$$

Also, the corresponding a is seen to be

$$a = (3/2)\log(\log(M))/\sqrt{2\log(M)} + \tilde{a},$$

where,

$$\tilde{a} = \frac{2\log\left[\mathsf{snr}\,(1 + 1/\mathcal{C})/((\pi)^{.25})\right]}{\sqrt{2\log(M)}}.$$

Express R_{total} in the form $\mathcal{C}/(1 + drop)$. Then with the above choices of r and a, and the bound on r_1 given in (61), one sees that $drop$ can be approximated by

$$\frac{3\log\log M + 4\log(\omega_1\,\mathsf{snr}) + 1/(4\mathcal{C}) + 3.35}{2\log M} + \frac{1 + 1/(2\mathcal{C})}{\sqrt{\pi\,\log M}},$$

where $\omega_1 = 1 + 1/\mathcal{C}$.

We remark that the above explicit expressions are given to highlight the nature of dependence of the rate drop on snr and M. These are quite conservative. For more accurate numerical evaluation see section VI on computational illustrations.

E. Definition of $\mathcal{C}^*$ and Proof of Theorem 2

In the previous subsection we gave the value of r and a that maximized, in an approximate sense, the outer code rate for given snr and M values and for large L. This led to explicit expressions for the maximal achievable outer code rate as a function of snr and M. We define $\mathcal{C}^*$ to be the inner code rate corresponding to this maximum achievable outer code rate. Thus,

$$\mathcal{C}^* = \frac{\mathcal{C}}{(1+\delta_a)^2\,[1 + r^*/\log M]}.$$

Similar to above, $\mathcal{C}^*$ can be written a $\mathcal{C}/(1 + drop^*)$ where $drop^*$ can be approximated by

$$\frac{3\log\log M + 4\log(\omega_1\mathsf{snr}) + 4/\omega_1 - 2}{2\log M} + \frac{1}{\sqrt{\pi\,\log M}},$$

with $\omega_1 = 1 + 1/\mathcal{C}$. We now give a proof of our main result.

Proof of Theorem 2: Take $r = r^* + \kappa$. Using

$$(1 + \kappa/\log M)(1 + r^*/\log M) \geq (1 + r/\log M),$$

we find that for the rate R as in Theorem 2, gap is at least $(r - r_1)/(\mathsf{snr}\log M)$ for $x \leq x_r$, with $x_r = r/(\mathsf{snr}\log M)$.

Take $f^* = (1/8)(r^* - r_1)^2/(\mathsf{snr}\log M)^2$, so that a is the same as given in the previous subsection. Now, we need to select $\rho > 1$ and $\eta > 0$, so that

$$f = \rho f^* \leq (\mathsf{gap} - \eta)^2/8 - 1/(2L_\pi).$$

Take $\omega = (1 + 1/\mathcal{C})/2$, so that $r^* = r_1 + 1/\omega$. One sees that we can satisfy the above requirement by taking η as $(1/2)\kappa/(\mathsf{snr}\log M)$ and $\rho = (1 + \kappa\omega/2)^2 - \epsilon_L$

$$\epsilon_L = \frac{(2\omega\mathsf{snr}\log M)^2}{L_\pi},$$

is of order $(\log M)^2/L$, and hence is negligible compared to the first term in ρ. Since it has little effect on the error exponent, for ease of exposition, we ignore this term. We also assume that $f = (\mathsf{gap} - \eta)^2/8$, ignoring the $1/(2L_\pi)$ term.

We select

$$h = \frac{\kappa}{(2\log M)^{3/2}}.$$

The fraction of mistakes,

$$\delta_{mis} = \frac{\mathsf{snr}}{2\mathcal{C}}\left[\frac{r}{\mathsf{snr}\log M} - (\mathsf{gap} - \eta)/2\right]$$

is calculated as in the previous subsection, except here we have to account for the positive η. Substituting the expression for gap and η gives the expression for δ_{mis} as in the theorem.

Next, let's look at the error probability. The error probability is given by

$$me^{-2L_\pi\eta^2 + mc_0} + me^{-L_\pi f D(\rho)/\rho} + me^{mh}e^{-nh^2/2}.$$

Notice that $nh^2/2$ is at least $(L_\pi\log M)h^2/(2\mathcal{C}^*)$, where we use that $L \geq L_\pi$ and $R \leq \mathcal{C}^*$. Thus the above probability is less than

$$\kappa_1\exp\{-L_\pi\min\{2\eta^2,\ f^*D(\rho),\ h^2\log M/(2\mathcal{C}^*)\}\}$$

with

$$\kappa_1 = 3m\,e^{m\max\{c_0,1/2\}},$$

where for the above we use $h < 1$.

Substituting, we see that $2\eta^2$ is $(1/2)\kappa^2/(\mathsf{snr}\log M)^2$ and $h^2\log M/(2\mathcal{C}^*)$ is

$$\frac{1}{16\mathcal{C}^*}\frac{\kappa^2}{(\log M)^2}.$$

Also, one sees that $D(\rho)$ is at least $2(\sqrt{\rho} - 1)^2/\rho$. Thus the term $f^*D(\rho)$ is at least

$$\frac{\kappa^2}{(4\mathsf{snr}\log M)^2(1 + \kappa\omega/2)}.$$

We bound from below the above quantity by considering two cases viz. $\kappa \leq 2/\omega$ and $\kappa > 2/\omega$. For the first case we have $1 + \kappa\omega/2 \leq 2$, so this quantity is bounded from below by $(1/2)\kappa^2/(4\mathsf{snr}\log M)^2$. For the second case use $\kappa/(1+\kappa\omega/2)$ is bounded from below by $1/\omega$, to get that this term is at least $(1/\omega)\kappa/(4\mathsf{snr}\log M)^2$.

Now we bound from below the quantity $\min\{2\eta^2,\ f^*D(\rho), h^2\log M/(2\mathcal{C}^*)\}$ appearing in the exponent. For $\kappa \leq 2/\omega$ this quantity is bounded from below by

$$\kappa_3\frac{\kappa^2}{(\log M)^2},$$

where

$$\kappa_3 = \min\left\{1/(32\mathsf{snr}^2),\ 1/(16\mathcal{C}^*)\right\}.$$

For $\kappa > 2/\omega$ this is quantity is at least

$$\min\left\{\kappa_3\frac{\kappa^2}{(\log M)^2}, \kappa_4\frac{\kappa}{\log M}\right\},$$

with

$$\kappa_4 = \frac{1}{8(1+1/\mathcal{C})\mathsf{snr}^2 \log M}.$$

Also notice that $\mathcal{C}^* - R$ is at most $\mathcal{C}^*\kappa/\log M$. Thus we have that

$$\min\{2\eta^2,\, f^* D(\rho),\, h^2 \log M/(2\mathcal{C}^*)\}$$

is at least

$$\min\left\{\kappa_3(\Delta^*)^2,\, \kappa_4 \Delta^*\right\}.$$

Further, recalling that $L_\pi = L\nu/(2\mathcal{C})$, we get that $\kappa_2 = \nu/(2\mathcal{C})$, which is near $\nu/(2\mathcal{C})$.

Regarding the value of m, recall that m is at most $2/(\mathsf{gap} - \eta)$. Using the above we get that m is at most $(2\omega\mathsf{snr})\log M$. Thus ignoring the $3m$, term $\kappa_1 = \kappa_{1,M}$ is polynomial in M with power $2\omega\mathsf{snr}\max\{c_0, 1/2\}$.

Part II is essentially the same as the use of Reed-Solomon codes in section VI of our companion paper [27].

In the proof of Proposition 1, we let ζ_i, for integer i, be constants that do not depend on L, M or n.

Proof of Proposition 1: Recall $R_{tot} = (1 - \delta_{mis})R$. Using the form of δ_{mis} and $\mathcal{C}^*$ for Proposition 2, one sees that R_{tot} may be expressed as,

$$R_{tot} = \left(1 - \zeta_1\delta_M - \zeta_2\frac{\kappa}{\log M}\right)\mathcal{C}. \tag{63}$$

Notice that M needs to be at least $\exp\{\zeta_3/\Delta^2\}$, where $\Delta = (\mathcal{C} - R_{tot})/\mathcal{C}$, for above to be satisfied. For a given section size M, the size of κ would be larger for a larger $\mathcal{C} - R_{tot}$. Choose κ so that $\zeta_2\kappa/\log M$ is at least $\zeta_1\delta_M$, so that by (63), one has,

$$\Delta \geq 2\zeta_2\frac{\kappa}{\log M}. \tag{64}$$

Now following the proof of Theorem 2, since the error exponent is of the form $\mathsf{const}\min\{\kappa/\log M, (\kappa/\log M)^2\}$, one sees that it is of the form $\mathsf{const}\,\Delta^2$ from (64).

VIII. Discussion

This paper demonstrated that the sparse superposition coding scheme, with the adaptive successive decoder and outer Reed-Solomon code, allows one to communicate at any rate below capacity, with block error probability that is exponentially small in L. It is shown in [8] that this exponent can be improved by a factor of $\sqrt{\log M}$ from using a Bernstein bound on the probability of the large deviation events analyzed here.

For fixed section size M, the power allocation (2) analyzed in this paper, allows one to achieve any R that is at least a drop of $1/\sqrt{\log M}$ of $\mathcal{C}$. In contrast, constant power allocation allows us to achieve rates up to a threshold rate $R_0 = 0.5\mathsf{snr}/(1+\mathsf{snr})$, which is bounded by $1/2$, but is near $\mathcal{C}$ for small snr. In [8] and [25] the alternative power allocation (5) is shown to allow for rates that is of order $\log\log M/\log M$ from capacity. Our experience shows that it is advantageous to use different power allocation schemes depending on the regime for snr. When snr is small, constant power allocation works better. The power allocation with leveling (5) works better for moderately large snr, whereas (2) is appropriate for larger snr values.

One of the requirements of the algorithm, as seen in the proof of Proposition 1, is that for fixed rate R_{tot}, the section size M is needed to be exponential in $1/\Delta^2$, using power allocation (2). Here Δ is the rate drop from capacity. Similar results hold for the other power allocations as well. However, this was not the case for the optimal ML–decoder, as seen in [27]. Consequently, it is still an open question whether there are practical decoders for the sparse superposition coding scheme which do not have this requirement on the dictionary size.

Appendix A
Proof of Lemma 4

For each $k \geq 2$, express X as,

$$X = \frac{G_1}{\|G_1\|}\mathcal{Z}_1^{\mathrm{T}} + \cdots + \frac{G_{k-1}}{\|G_{k-1}\|}\mathcal{Z}_{k-1}^{\mathrm{T}} + \xi_k V_k,$$

where $\xi_k = [\xi_{k,k} : \ldots : \xi_{k,n}]$ is an $n \times (n-k+1)$ orthonormal matrix, with columns $\xi_{k,i}$, for $i = k, \ldots, n$, being orthogonal to $G_1, \ldots, G_{k-1}$. There is flexibility in the choice of the $\xi_{k,i}$'s, the only requirement being that they depend on only $G_1, \ldots, G_{k-1}$ and no other random quantities. For convenience, we take these $\xi_{k,i}$'s to come from the Gram-Schmidt orthogonalization of $G_1, \ldots, G_{k-1}$ and the columns of the identity matrix.

The matrix V_k, which is $(n-k+1) \times N$ dimensional, is also denoted as,

$$V_k = [V_{k,1} : V_{k,2} : \ldots : V_{k,N}].$$

The columns $V_{k,j}$, where $j = 1, \ldots, N$ gives the coefficients of the expansion of the column X_j in the basis $\xi_{k,k}, \xi_{k,k+1}, \ldots, \xi_{k,n}$. We also denote the entries of V_k as $V_{k,i,j}$, where $i = k, \ldots, n$ and $j = 1, \ldots, N$.

We prove that conditional on $\mathcal{F}_{k-1}$, the distribution of $(V_{k,i,j} : j \in J_{k-1})$, for $i = k, \ldots, n$, is i.i.d. Normal Normal$(0, \Sigma_{k-1})$. The proof is by induction.

The stated property is true initially, at $k = 2$, from Lemma 3. Recall that the rows of the matrix U in Lemma 3 are i.i.d. Normal$(0, \Sigma_1)$. Correspondingly, since $V_2 = \xi_2^{\mathrm{T}} U$, and since the columns of ξ_2 are orthonormal, and independent of U, one gets that the rows of V_2 are i.i.d Normal$(0, \Sigma_1)$ as well.

Presuming the stated conditional distribution property to be true at k, we conduct analysis, from which its validity will be demonstrated at $k+1$. Along the way the conditional distribution properties of G_k, $N_{k,j}$, and $\mathcal{Z}_{k,j}$ are obtained as consequences. As for $\hat{w}_k$ and δ_k we first obtain them by explicit recursions and then verify the stated form.

Denote as

$$G_{k,i}^{coef} = -\sum_{j \in \mathsf{dec}_{k-1}} \sqrt{P_j}\, V_{k,i,j} \quad \text{for } i = k, \ldots, n\ . \tag{65}$$

Also denote as,

$$G_k^{coef} = (G_{k,k}^{coef},\, G_{k,k+1}^{coef}, \ldots, G_{k,n}^{coef})^{\mathrm{T}}.$$

The vector G_k^{coef} gives the representation of G_k in the basis consisting of columns vectors of ξ_k. In other words, $G_k = \xi_k G_k^{coef}$.

Notice that,

$$\mathcal{Z}_{k,j} = V_{k,j}^{\mathrm{T}} G_k^{coef} / \|G_k^{coef}\|. \tag{66}$$

Further, since $V_{k,j}$ and G_k^{coef} are jointly normal conditional on $\mathcal{F}_{k-1}$, one gets, through conditioning on G_k^{coef} that,

$$V_{k,j} = b_{k-1,j}\, G_k^{coef}/\sigma_k + U_{k,j}.$$

Denote as $U_k = [U_{k,1} : U_{k,2} : \ldots : U_{k,N}]$, which is an $(n-k+1) \times N$ dimensional matrix like V_k. The entries of U_k are denoted as $U_{k,i,j}$, where $i = k, \ldots, n$ and $j = 1, \ldots, N$. The matrix U_k is independent of G_k^{coef}, conditioned on $\mathcal{F}_{k-1}$. Further, from the representation (66), one gets that

$$\mathcal{Z}_{k,j} \;=\; b_{k-1,j}\, \|G_k^{coef}\|/\sigma_k \;+\; N_{k,j}, \tag{67}$$

with,

$$N_{k,j} = U_{k,j}^{\mathrm{T}} G_k^{coef} / \|G_k^{coef}\|.$$

For the conditional distribution of $G_{k,i}^{coef}$ given $\mathcal{F}_{k-1}$, independence across i, conditional normality and conditional mean 0 are properties inherited from the corresponding properties of the $V_{k,i,j}$. To obtain the conditional variance of $G_{k,i}^{coef}$, given by (65), use the conditional covariance

$$\Sigma_{k-1} = I - \delta_{k-1}\delta_{k-1}^T$$

of $(V_{k,i,j} : j \in J_{k-1})$. The identity part contributes $\sum_{j \in \mathsf{dec}_{k-1}} P_j$ which is $(\hat{q}_{k-1} + \hat{f}_{k-1})P$; whereas, the $\delta_{k-1}\delta_{k-1}^T$ part, using the presumed form of δ_{k-1}, contributes an amount seen to equal $\nu_{k-1}[\sum_{j \in sent \cap \mathsf{dec}_{k-1}} P_j/P]^2\, P$ which is $\nu_{k-1}\hat{q}_{k-1}^2 P$. It follows that the conditional expected square for $G_{k,i}^{coef}$, for $i = k, \ldots, n$ is

$$\sigma_k^2 \;= [\,\hat{q}_{k-1} + \hat{f}_{k-1} - \hat{q}_{k-1}^2\, \nu_{k-1}]\, P.$$

Conditional on $\mathcal{F}_{k-1}$, the distribution of

$$\|G_k^{coef}\|^2 = \sum_{i=k}^{n} (G_{k,i}^{coef})^2$$

is that of $\sigma_k^2\, \mathcal{X}_{n-k+1}^2$, a multiple of a Chi-square with $n-k+1$ degrees of freedom.

Next we compute $b_{k-1,j}$ in (67), which is the value of

$$\mathbb{E}[V_{k,i,j}\, G_{k,i}^{coef} | \mathcal{F}_{k-1}]/\sigma_k$$

for any of the coordinates $i = k, \ldots, n$. Consider the product $V_{k,i,j}\, G_{k,i}^{coef}$ in the numerator. Using the representation of $G_{k,i}^{coef}$ in (65), one has $\mathbb{E}[V_{k,i,j} G_{k,i}^{coef} | \mathcal{F}_{k-1}]$ is

$$-\sum_{j' \in \mathsf{dec}_{k-1}} \sqrt{P_{j'}}\big[1_{j'=j} - \delta_{k-1,j}\delta_{k-1,j'}\big],$$

which simplifies to $-\sqrt{P_j}\,\big[1_{j \in \mathsf{dec}_{k-1}} - \nu_{k-1}\hat{q}_{k-1} 1_{j\, sent}\big]$. So for j in $J_k = J_{k-1} - \mathsf{dec}_{k-1}$, we have the simplification

$$b_{k-1,j} = \frac{\hat{q}_{k-1}\, \nu_{k-1} \beta_j}{\sigma_k}. \tag{68}$$

Also, for j, j' in J_k, the product takes the form

$$b_{k-1,j} b_{k-1,j'} = \delta_{k-1,j}\delta_{k-1,j'} \frac{\hat{q}_{k-1}\nu_{k-1}}{1 + \hat{f}_{k-1}/\hat{q}_{k-1} - \hat{q}_{k-1}\nu_{k-1}}.$$

Here the ratio simplifies to $\hat{q}_{k-1}^{adj}\nu_{k-1}/(1 - \hat{q}_{k-1}^{adj}\nu_{k-1})$.

Now determine the features of the joint normal distribution of the

$$U_{k,i,j} = V_{k,i,j} - b_{k-1,j}\, G_{k,i}^{coef}/\sigma_k,$$

for $j \in J_k$, given $\mathcal{F}_{k-1}$. Given $\mathcal{F}_{k-1}$, the $(U_{k,i,j} : j \in J_k)$ are i.i.d across choices of i, but there is covariance across choices of j for fixed i. This conditional covariance $\mathbb{E}[U_{k,i,j} U_{k,i,j'} | \mathcal{F}_{k-1}]$, by the choice of $b_{k-1,j}$, reduces to $\mathbb{E}[V_{k,i,j} V_{k,i,j'} | \mathcal{F}_{k-1}] - b_{k-1,j} b_{k-1,j'}$ which, for $j \in J_k$, is

$$1_{j=j'} - \delta_{k-1,j}\delta_{k-1,j'} - b_{k-1,j} b_{k-1,j'}.$$

That is, for each i, the $(U_{k,i,j} : j \in J_k)$ have the joint $\mathrm{Normal}_{J_k}(0, \Sigma_k)$ distribution, conditional on $\mathcal{F}_{k-1}$, where Σ_k again takes the form $1_{j,j'} - \delta_{k,j}\delta_{k,j'}$ where

$$\delta_{k,j}\delta_{k,j'} \;=\; \delta_{k-1,j}\delta_{k-1,j'} \left\{ 1 \;+\; \frac{\hat{q}_{k-1}^{adj}\, \nu_{k-1}}{1 - \hat{q}_{k-1}^{adj}\nu_{k-1}} \right\},$$

for j, j' now restricted to J_k. The quantity in braces simplifies to $1/(1 - \hat{q}_{k-1}^{adj}\nu_{k-1})$. Correspondingly, the recursive update rule for ν_k is

$$\nu_k \;=\; \frac{\nu_{k-1}}{1 \;-\; \hat{q}_{k-1}^{adj}\, \nu_{k-1}}.$$

Consequently, the joint distribution for $(N_{k,j} : j \in J_k)$ is determined, conditional on $\mathcal{F}_{k-1}$. It is also the normal Normal$(0, \Sigma_k)$ distribution and $(N_{k,j} : j \in J_k)$ is conditionally independent of the coefficients of G_k^{coef}, given $\mathcal{F}_{k-1}$. After all, the

$$N_{k,j} = U_{k,j}^{\mathrm{T}} G_k^{coef} / \|G_k^{coef}\|$$

have this $\mathrm{Normal}_{J_k}(0, \Sigma_k)$ distribution, conditional on G_k^{coef} and $\mathcal{F}_{k-1}$, but since this distribution does not depend on G_k^{coef} we have the stated conditional independence.

This makes the conditional distribution of the $\mathcal{Z}_{k,j}$, given $\mathcal{F}_{k-1}$, as given in (67), a location mixture of normals with distribution of the shift of location determined by the Chi-square distribution of $\mathcal{X}_{n-k+1}^2 = \|G_k^{coef}\|^2/\sigma_k^2$. Using the form of $b_{k-1,j}$, for j in J_k, the location shift $b_{k-1,j}\, \mathcal{X}_{n-k+1}$ may be written

$$\sqrt{\hat{w}_k\, C_{j,R}}\,\big[\, \mathcal{X}_{n-k+1}/\sqrt{n}\,\big]\, 1_{j\, sent},$$

where

$$\hat{w}_k = \frac{n\, b_{k,j}^2}{C_{j,R}}.$$

The numerator and denominator has dependence on j through P_j, so canceling the P_j produces a value for $\hat{w}_k$. Indeed, $C_{j,R} = (P_j/P)\nu(L/R)\log M$ equals $n(P_j/P)\nu$ and $b_{k-1,j}^2 = P_j \hat{q}_{k-1}^{adj}\, \nu_{k-1}^2/[1 - \hat{q}_{k-1}^{adj}\nu_{k-1}]$. So this $\hat{w}_k$ may be expressed as

$$\hat{w}_k \;=\; \frac{\nu_{k-1}}{\nu} \;\frac{\hat{q}_{k-1}^{adj}\, \nu_{k-1}}{1 - \hat{q}_{k-1}^{adj}\nu_{k-1}},$$

which, using the update rule for ν_{k-1}, is seen to equal

$$\hat{w}_k = \frac{\nu_{k-1} - \nu_k}{\nu}.$$

Further, repeatedly apply $\nu_{k'}/\nu_{k'-1} = 1/(1-\hat{q}^{adj}_{k'-1}\,\nu_{k'-1})$, for k' from k to 2, each time substituting the required expression on the right and simplifying to obtain

$$\frac{\nu_k}{\nu_{k-1}} \;=\; \frac{1 \,-\, (\hat{q}^{adj}_1 + \cdots + \hat{q}^{adj}_{k-2})\;\nu}{1 - (\hat{q}^{adj}_1 + \cdots + \hat{q}^{adj}_{k-2} + \hat{q}^{adj}_{k-1})\,\nu}.$$

This yields $\nu_k = \nu \hat{s}_k$, which, when plugged into the expressions for $\hat{w}_k$, establishes the form of $\hat{w}_k$ as given in the lemma.

We need to prove that conditional on $\mathcal{F}_k$ that the rows of V_{k+1}, for $j \in J_k$, are i.i.d. Normal$_{J_k}(0, \Sigma_k)$. Recall that $V_{k+1} = \xi^{\mathrm{T}}_{k+1} X$. Since the column span of ξ_{k+1} is contained in that of ξ_k, one may also write V_{k+1} as $\xi^{\mathrm{T}}_{k+1}\xi_k V_k$. Similar to the representation $G_k = \xi_k G^{coef}_k$, express the columns of ξ_{k+1} in terms of the columns of ξ_k as $\xi_{k+1} = \xi_k \xi^{coef}_k$, where ξ^{coeff}_k is an $(n-k+1)\times(n-k)$ dimensional matrix. Using this representation one gets that $V_{k+1} = (\xi^{coef}_k)^{\mathrm{T}} V_k$.

Notice that ξ_k is a function of $\mathcal{F}_{k-1}$ and that ξ_{k+1} is a function of $\{\mathcal{F}_{k-1}, G_k\}$. Correspondingly, ξ^{coef}_k is also a function of $\{\mathcal{F}_{k-1}, G_k\}$. Further, because of the orthonormality of ξ_k and ξ_{k+1}, one gets that the columns of ξ^{coef}_k are also orthonormal. Further, as G_k is orthonormal to ξ_{k+1}, one has that G^{coef}_k is orthogonal to the columns of ξ^{coef}_k as well.

Accordingly, one has that $V_{k+1} = (\xi^{coef}_k)^{\mathrm{T}} U_k$. Consequently, using the independence of U_k and G^{coef}_k, and the above, one gets that conditional on $\{\mathcal{F}_{k-1}, G_k\}$, for $J \in J_k$, the rows of V_{k+1} are i.i.d. Normal$_{J_k}(0, \Sigma_k)$.

We need to prove that conditional on $\mathcal{F}_k$, the distribution of V_{k+1} is as above. Recall that $\mathcal{F}_k$ is the set of functions, or more precisely, the σ-field, of random variables $\{\mathcal{F}_{k-1}, G_k, \mathcal{Z}_k\}$. Equivalently, it the set functions of $\{\mathcal{F}_{k-1}, G_k, N_k\}$. Consequently, the claim follows from the conclusion of the previous paragraph by noting that V_{k+1} is independent of $N_k = (G^{coef}_k)^{\mathrm{T}} U_k$, conditional on $\{\mathcal{F}_{k-1}, G_k\}$, as G^{coef}_k is orthogonal to ξ^{coeff}_k.

This completes the proof of the Lemma 4.

Appendix B
The Method of Nearby Measures

Let $b \in \mathbb{R}^n$, be such that $\|b\|^2 = \nu < 1$. Further, let $\mathbb{P}$ be the probability measure of a Normal$(0, \Sigma)$ random variable, where $\Sigma = I - bb^T$, and let $\mathbb{Q}$ be the measure of a Normal$_n(0, I)$ random variable. Then we have,

Lemma 12:

$$\mathbb{P}[A] \le \mathbb{Q}[A] e^{c_0},$$

where $c_0 = -(1/2)\log(1-\nu)$.

Proof: If $p(z)$, $q(z)$, denote the densities of the random variables with measures $\mathbb{P}$ and $\mathbb{Q}$ respectively, then $\max_z p(z)/q(z)$ equals $1/(1-\nu)^{1/2}$, which is also e^{c_0}. From the densities Normal$(0, I - bb^T)$ and Normal$(0, I)$ this claim can be established from noting that after an orthogonal transformation these measures are only different in one variable, which is either Normal$(0, 1-\nu)$ or Normal$(0, 1)$, for which the maximum ratio of the densities occurs at the origin and is simply the ratio of the normalizing constants.

Correspondingly,

$$\begin{aligned}\mathbb{P}(A) &= \int_A p(z)\,dz \\ &\le e^{c_0} \int_A q(z)\,dz = \mathbb{Q}(A) e^{c_0}.\end{aligned}$$

This completes the proof of the lemma. ∎

Proof of Lemma 5: We are to show that for events A determined by the random variables (32), the probability $\mathbb{P}[A]$ is not more than $\mathbb{Q}[A]e^{kc_0}$. Write the probability as an iterated expectation conditioning on $\mathcal{F}_{k-1}$. That is, $\mathbb{P}[A] = \mathbb{E}\left[\mathbb{P}[A|\mathcal{F}_{k-1}]\right]$. To determine membership in A, conditional on $\mathcal{F}_{k-1}$, we only need $N_{k,J_k} = (N_{k,j} : j \in J_k)$ where J_k is determined by $\mathcal{F}_{k-1}$. Thus

$$\mathbb{P}[A] = \mathbb{E}_{\mathbb{P}}\left[\mathbb{P}_{\mathcal{X}^2_{d_k}, N_{k,J_k}|\mathcal{F}_{k-1}}[A]\right],$$

where we use the subscript on the outer expectation to denote that it is with respect to $\mathbb{P}$ and the subscripts on the inner conditional probability to indicate the relevant variables. For this inner probability switch to the nearby measure $\mathbb{Q}_{\mathcal{X}_{d_k}, N_{k,J_k}|\mathcal{F}_{k-1}}$. These conditional measures agree concerning the distribution of the independent $\mathcal{X}^2_{d_k}$, so what matters is the ratio of the densities corresponding to $\mathbb{P}_{N_{k,J_k}|\mathcal{F}_{k-1}}$ and $\mathbb{Q}_{N_{k,J_k}|\mathcal{F}_{k-1}}$.

We claim that the ratio of these densities in bounded by e^{c_0}. To see this, recall that from Lemma 4 that $\mathbb{P}_{N_{k,J_k}|\mathcal{F}_{k-1}}$ is $N_{J_k}(0, \Sigma_k)$, with $\Sigma_k = I - \delta_k \delta_k^T$. Now

$$||\delta_k||^2 = \nu_k \sum_{j \in sent \cap J_k} P_j/P$$

which is $(1 - (\hat{q}_1 + \cdots + \hat{q}_{k-1}))\nu_k$. Noting that $\nu_k = \hat{s}_k \nu$ and $\hat{s}_k(1 - (\hat{q}_1 + \cdots + \hat{q}_{k-1}))$ is at most 1, we get that $||\delta_k||^2 \le \nu$.

So with the switch of conditional distribution, we obtain a bound with a multiplicative factor of e^{c_0}. The bound on the inner expectation is then a function of $\mathcal{F}_{k-1}$, so the conclusion follows by induction. This completes the proof of Lemma 5.

Appendix C
Proof of Lemma 6

For $k = 1$, the $\theta_1 = g(0) - \eta$ is at least $\mathsf{gap} - \eta$. Consider $\theta_k = g_L(q^{adj,tot}_{k-1}) - \eta$, for $k > 1$. Notice that

$$q^{adj,tot}_{k-1} \ge \sum_{k'=1}^{k-1} q_{k'} - (k-1) f,$$

using $q/(1 + f/q) \ge q - f$. Now, from the definition of q_k in (44), one has

$$\sum_{k'=1}^{k-1} q_{k'} = \theta_{k-1} - (k-1)(f + 1/L_\pi).$$

Consequently,

$$q^{adj,tot}_{k-1} \ge \theta_{k-1} - (k-1)(2f + 1/L_\pi). \tag{69}$$

Denote m as the first k for which $q_{k-1}^{adj,tot}$ exceeds x_r. For any $k < m$, as $q_{k-1}^{adj,tot} \leq x_r$, using the fact that g_L is accumulative till x_r, one gets that

$$\theta_k \geq q_{k-1}^{adj,tot} + \mathsf{gap} - \eta.$$

Accordingly, using (69), one gets that

$$\theta_k \geq \theta_{k-1} - (k-1)(2f + 1/L_\pi) + \mathsf{gap} - \eta, \qquad (70)$$

or in other words, for $k < m$, one has

$$\theta_k - \theta_{k-1} \geq -m(2f + 1/L_\pi) + \mathsf{gap} - \eta.$$

We want to arrange the difference $\theta_k - \theta_{k-1}$ to be at least a positive quantity which we denote by Λ. Notice that this gives $m \leq 1/\Lambda$, since the θ_k's are bounded by 1. Correspondingly, we solve for Λ in,

$$\Lambda = -(1/\Lambda)(2f + 1/L_\pi) + \mathsf{gap} - \eta,$$

and see that the solution is

$$\Lambda = \frac{(\mathsf{gap} - \eta)}{2}\left[1 + \left(1 - 4\frac{(2f + 1/L_\pi)}{(\mathsf{gap} - \eta)^2}\right)^{1/2}\right], \qquad (71)$$

which is well defined since f satisfies (45). Also notice that from (71) that $\Lambda \geq (\mathsf{gap} - \eta)/2$, making $m \leq 2/(\mathsf{gap} - \eta)$. Also $\theta_m = g_L(q_{m-1}^{adj,tot}) - \eta$, which is at least $g_L(x_r) - \eta$ since g_L is increasing. The latter quantity is at least $x_r + \mathsf{gap} - \eta$.

Appendix D
Proof of Lemma 7

We prove the lemma by first showing that

$$A_{1,m} \subseteq \tilde{A}_{1,m} \cup B_{1,m} \cup D_{1,m}. \qquad (72)$$

Next, we prove that $B_{1,m}$ is contained in $\tilde{B}_{1,m}$. This will prove the lemma.

We start by showing (72). We first show that on the set

$$\{\hat{\theta}_k \geq \theta_k\} \cap E_{k-1}^c \cap D_k^c, \qquad (73)$$

condition (18), that is,

$$\sum_{j \in \mathsf{Dec}_{k-1}} \pi_j + \sum_{j \in J - \mathsf{Dec}_{k-1}} \pi_j 1_{\{\mathcal{Z}_{k,j}^{comb} \geq \tau\}} \geq \theta_k, \qquad (74)$$

is satisfied. Following the arguments of subsection II-B regarding pacing the steps, this will ensure that the size of the decoded set after k steps, that is size_k, is near θ_k, or more precisely

$$\theta_k - 1/L_\pi < \mathsf{size}_k \leq \theta_k, \qquad (75)$$

as given in (17).

Notice that the left side of (74) is at least

$$\sum_{j \in \mathsf{sent}} \pi_j 1_{\{\mathcal{Z}_{k,j}^{comb} \geq \tau\}},$$

since the sum in (74) is over all terms in j, including those in sent, and further, for each term j, the contribution to the sum is at least $\pi_j 1_{\{\mathcal{Z}_{k,j}^{comb} \geq \tau\}}$.

Further, using the fact that

$$H_{k,j} \subseteq \{\mathcal{Z}_{k,j}^{comb} \geq \tau\} \quad \text{on } E_{k-1}^c \cap D_k^c$$

from (37), one gets that,

$$\sum_{j \in \mathsf{sent}} \pi_j 1_{\{\mathcal{Z}_{k,j}^{comb} \geq \tau\}} \geq \hat{\theta}_k \quad \text{on } E_{k-1}^c \cap D_k^c \ .$$

Correspondingly, on the set (73) the inequality (74), and consequently the relation (75) also holds.

Next, for each k, denote

$$\tilde{E}_k = \tilde{A}_{1,k} \cup B_{1,k} \cup D_{1,k}. \qquad (76)$$

We claim that for each $k = 1, \ldots, m$, one has

$$\tilde{E}_k^c \subseteq A_{1,k}^c.$$

We prove the claim through induction on k. Notice that the claim for $k = m$ is precisely statement (72). Also, the claim implies that $\tilde{E}_k^c \subseteq E_k^c$, for each k, where recall that $E_k = A_{1,k} \cup B_{1,k} \cup D_{1,k}$.

We first prove the claim for $k = 1$. We see that,

$$\tilde{E}_1^c = \{\hat{\theta}_1 \geq \theta_1\} \cap \{\hat{f}_1 \leq f\} \cap D_1^c.$$

Using the arguments above, we see that on $\{\hat{\theta}_1 \geq \theta_1\} \cap D_1^c$, the relation $\theta_1 - 1/L_\pi < \mathsf{size}_1$ holds. Now, since $\mathsf{size}_1 = \hat{q}_1 + \hat{f}_1$, one gets that

$$\hat{q}_1 \geq \theta_1 - \hat{f}_1 - 1/L_\pi \quad \text{on } \{\hat{\theta}_1 \geq \theta_1\} \cap D_1^c.$$

The right side of the aforementioned inequality is at least q_1 on $\tilde{E}_1^c$, using $\hat{f}_1 \leq f$. Consequently, the claim is proved for $k = 1$.

Assume that the claim holds till $k-1$, that is, assume that $\tilde{E}_{k-1}^c \subseteq A_{1,k-1}^c$. We now prove that $\tilde{E}_k^c \subseteq A_{1,k}^c$ as well. Notice that

$$\tilde{E}_k^c = \{\hat{\theta}_k \geq \theta_k\} \cap \tilde{E}_{k-1}^c \cap D_k^c \cap \{\hat{f}_k \leq f\},$$

which, using $\tilde{E}_{k-1}^c \subseteq E_{k-1}^c$ from the induction hypothesis, one gets that

$$\theta_k - 1/L_\pi < \mathsf{size}_k \quad \text{on } \{\hat{\theta}_k \geq \theta_k\} \cap \tilde{E}_{k-1}^c \cap D_k^c.$$

Accordingly,

$$\theta_k - 1/L_\pi < \mathsf{size}_k \leq \theta_k \quad \text{on } \tilde{E}_k^c.$$

Further, as $\tilde{E}_k^c$ is contained in $\tilde{E}_{k-1}^c$, one gets that

$$\theta_{k-1} - 1/L_\pi < \mathsf{size}_{k-1} \leq \theta_{k-1} \quad \text{on } \tilde{E}_k^c.$$

Consequently, combining the above, one has

$$\begin{aligned}\mathsf{size}_k - \mathsf{size}_{k-1} &= \hat{q}_k + \hat{f}_k \\ &\geq \theta_k - \theta_{k-1} - f - 1/L_\pi \quad \text{on } \tilde{E}_k^c.\end{aligned}$$

Consequently, on $\tilde{E}_k^c$, we have $\hat{q}_k \geq q_k$, using the expression for q_k given in (44), and the fact that $f_k \leq f$ on $\tilde{E}_k^c$. Combining this with the fact that $\tilde{E}_k^c$ is contained in $A_{1,k-1}^c$, since $\tilde{E}_k^c \subseteq \tilde{E}_{k-1}^c$ and $\tilde{E}_{k-1}^c \subseteq A_{1,k-1}^c$ from the induction hypothesis, one gets that $\tilde{E}_k^c \subseteq A_{1,k}^c$.

This proves the induction hypothesis. In particular, it holds for $k = m$, which, taking complements, proves the statement (72).

Next, we show that $B_{1,m} \subseteq \tilde{B}_{1,m}$. This is straightforward since recall that from subsection IV-E that one has $\hat{f}_k \leq \hat{f}_k^{up}$, for each k. Correspondingly, $B_{1,m}$ is contained in $\tilde{B}_{1,m}$.

Consequently, from (72) and the fact that $B_{1,m} \subseteq \tilde{B}_{1,m}$, one gets that E_m is contained in $\tilde{A}_{1,m} \cup \tilde{B}_{1,m} \cup D_{1,m}$. This proves the lemma.

APPENDIX E
TAILS FOR WEIGHTED BERNOULLI SUMS

Lemma 13: Let W_j, $1 \le j \le N$ be N independent Bernoulli(r_j) random variables. Furthermore, let α_j, $1 \le j \le N$ be non-negative weights that sum to 1 and let $N_\alpha = 1/\max_j \alpha_j$. Then the weighted sum $\hat{r} = \sum_j \alpha_j W_j$ which has mean given by $r^* = \sum_j \alpha_j r_j$, satisfies the following large deviation inequalities. For any r with $0 < r < r^*$,

$$P(\hat{r} < r) \le \exp\left\{-N_\alpha D(r\|r^*)\right\}$$

and for any $\tilde{r}$ with $r^* < \tilde{r} < 1$,

$$P(\hat{r} > \tilde{r}) \le \exp\left\{-N_\alpha D(\tilde{r}\|r^*)\right\}$$

where $D(r\|r^*)$ denotes the relative entropy between Bernoulli random variables of success parameters r and r^*.

Proof of Lemma 13: Let's prove the first part. The proof of the second part is similar.

Denote the event

$$\mathcal{A} = \{\underline{W} : \sum_j \alpha_j W_j \le r\}$$

with $\underline{W}$ denoting the N-vector of W_j's. Proceeding as in Csiszár [18] we have that

$$\begin{aligned} P(\mathcal{A}) &= \exp\{-D\big(P_{\underline{W}|\mathcal{A}}\|P_{\underline{W}}\big)\} \\ &\le \exp\Big\{-\sum_j D\big(P_{W_j|\mathcal{A}}||P_{W_j}\big)\Big\} \end{aligned}$$

Here $P_{\underline{W}|\mathcal{A}}$ denotes the conditional distribution of the vector $\underline{W}$ conditional on the event $\mathcal{A}$ and $P_{W_j|\mathcal{A}}$ denotes the associated marginal distribution of W_j conditioned on $\mathcal{A}$. Now

$$\sum_j D\big(P_{W_j|\mathcal{A}}\|P_{W_j}\big) \ge N_\alpha \sum_j \alpha_j D\big(P_{W_j|\mathcal{A}}\|P_{W_j}\big).$$

Furthermore, the convexity of the relative entropy implies that

$$\sum_j \alpha_j D(P_{W_j|\mathcal{A}} \parallel P_{W_j}) \ge D\left(\sum_j \alpha_j P_{W_j|\mathcal{A}} \parallel \sum_j \alpha_j P_{W_j}\right).$$

The sums on the right denote α mixtures of distributions $P_{W_j|\mathcal{A}}$ and P_{W_j}, respectively, which are distributions on $\{0, 1\}$, and hence these mixtures are also distributions on $\{0, 1\}$. In particular, $\sum_j \alpha_j P_{W_j}$ is the Bernoulli(r^*) distribution and $\sum_j \alpha_j P_{W_j|\mathcal{A}}$ is the Bernoulli(r_e) distribution where

$$r_e = \mathrm{E}\big[\sum_j \alpha_j W_j \,\big|\, \mathcal{A}\big] = \mathrm{E}\big[\hat{r} \,\big|\, \mathcal{A}\big].$$

But in the event $\mathcal{A}$ we have $\hat{r} \le r$ so it follows that $r_e \le r$. As $r < r^*$ this yields $D(r_e \parallel r^*) \ge D(r \parallel r^*)$. This completes the proof of Lemma 13.

APPENDIX F
LOWER BOUNDS ON D

Lemma 14: For $p \ge p^*$, the relative entropy between Bernoulli(p) and Bernoulli(p^*) distributions has the succession of lower bounds

$$D_{Ber}(p\|p^*) \ge D_{Poi}(p\|p^*) \ge 2\big(\sqrt{p} - \sqrt{p^*}\big)^2 \ge \frac{(p-p^*)^2}{2p}$$

where $D_{Poi}(p\|p^*) = p\log p/p^* + p^* - p$ is also recognizable as the relative entropy between Poisson distributions of mean p and p^* respectively.

Proof: The Bernoulli relative entropy may be expressed as the sum of two positive terms, one of which is $p\log p/p^* + p^* - p$, and the other is the corresponding term with $1-p$ and $1-p^*$ in place of p and p^*, so this demonstrates the first inequality. Now suppose $p > p^*$. Write $p\log p/p^* + p^* - p$ as $p^*F(s)$ where $F(s) = 2s^2\log s + 1 - s^2$ with $s^2 = p/p^*$ which is at least 1. This function F and its first derivative $F'(s) = 4s\log s$ have value equal to 0 at $s = 1$, and its second derivative $F''(s) = 4 + 4\log s$ is at least 4 for $s \ge 1$. So by second order Taylor expansion $F(s) \ge 2(s-1)^2$ for $s \ge 1$. Thus $p\log p/p^* + p^* - p$ is at least $2\big(\sqrt{p} - \sqrt{p^*}\big)^2$. Furthermore $2(s-1)^2 \ge (s^2-1)^2/(2s^2)$ as, taking the square root of both sides, it is seen to be equivalent to $2(s-1) \ge s^2 - 1$, which, factoring out $s-1$ from both sides, is seen to hold for $s \ge 1$. From this we have the final lower bound $(p-p^*)^2/(2p)$. ∎

ACKNOWLEDGMENT

We thank Dan Spielman, Edmund Yeh, Mokshay Madiman and Imre Teletar for helpful conversations. We thank David Smalling who completed a number of simulations of earlier incarnations of the decoding algorithm for his Yale applied math senior project in spring term of 2009 and Yale statistics masters student Creighton Hauikulani who took the simulations further in 2009 and 2010.

REFERENCES

[1] A. Abbe and A. R. Barron, "Polar codes for the AWGN," in *Proc. IEEE ISIT*, Aug. 2011, pp. 194–198.
[2] M. Akçakaya and V. Tarokh, "Shannon-theoretic limits on noisy compressive sampling," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 492–504, Jan. 2010.
[3] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.
[4] E. Arikan and E. Telatar, "On the rate of channel polarization," in *Proc. IEEE ISIT*, Apr. 2009, pp. 1493–1495.
[5] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, May 1993.
[6] A. R. Barron and S. Cho, "High-rate sparse superposition codes with iteratively optimal estimates," in *Proc. IEEE ISIT*, Jul. 2012, pp. 120–124.
[7] A. R. Barron, A. Cohen, W. Dahmen, and R. A. DeVore, "Approximation and learning by greedy algorithms," *Ann. Statist.*, vol. 36, no. 1, pp. 64–94, 2008.
[8] A. R. Barron and A. Joseph, "Toward fast reliable communication at rates near capacity with Gaussian noise," in *Proc. IEEE ISIT*, Jun. 2010, pp. 315–319.
[9] A. R. Barron and A. Joseph, "Analysis of fast sparse superposition codes," in *Proc. IEEE ISIT*, Aug. 2011, pp. 1772–1776.

[10] A. R. Barron and A. Joseph, "Sparse superposition codes are fast and reliable at rates approaching capacity with Gaussian noise," Dept. Statist., Yale Univ., New Haven, CT, USA, Tech. Rep., 2011.
[11] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.
[12] M. Bayati and A. Montanari, "The LASSO risk for Gaussian matrices," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 1997–2017, Apr. 2012.
[13] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proc. IEEE ICC*, vol. 2. May 1993, pp. 1064–1070.
[14] E. J. Candès and Y. Plan, "Near-ideal model selection by ł$_1$ minimization," *Ann. Statist.*, vol. 37, no. 5A, pp. 2145–2177, 2009.
[15] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
[16] T. Cover, "Broadcast channels," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 2–14, Jan. 1972.
[17] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, vol. 6. New York, NY, USA: Wiley, 1991.
[18] I. Csiszár, "Sanov property, generalized I-projection and a conditional limit theorem," *Ann. Probab.*, vol. 12, no. 3, pp. 768–793, 1984.
[19] A. K. Fletcher, V.K. Goyal, and S. Rangan, "A sparsity detection framework for on-off random access channels," in *Proc. IEEE ISIT*, Jul. 2009, pp. 169–173.
[20] A. K. Fletcher and S. Rangan, "Orthogonal matching pursuit: A Brownian motion analysis," *IEEE Trans. Signal Process.*, vol. 60, no. 3, pp. 1010–1021, Mar. 2012.
[21] G. David Forney, "Concatenated codes," DTIC, Fort Belvoir, VA, USA, Tech. Rep., 1965.
[22] R. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.
[23] C. Huang, G. H. L. Cheang, and A. R. Barron, "Risk of penalized least squares, greedy selection and L_1 penalization for flexible function libraries," Ph.D. dissertation, Dept. Statist., Yale Univ., New Haven, CT, USA, Nov. 2008.
[24] L. Jones, "A simple lemma for optimization in a Hilbert space, with application to projection pursuit and neural net training," *Ann. Statist.*, vol. 20, no. 1, pp. 608–613, Mar. 1992.
[25] A. Joseph, "Achieving information-theoretic limits with high-dimensional regression," Ph.D. dissertation, Yale Univ., New Haven, CT, USA, Jun. 2012.
[26] A. Joseph, "Variable selection in high dimensions with random designs and orthogonal matching pursuit," *J. Mach. Learn. Res.*, pp. 1771–1800, 2013.
[27] A. Joseph and A. R. Barron, "Least squares superposition coding of moderate dictionary size are reliable at rates up to channel capacity," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 2541–2557, May 2012.
[28] I. Kontoyiannis, S. Gitzenis, and K. R. Rad, "Superposition codes for Gaussian vector quantization," in *Proc. IEEE Inf. Theory Workshop*, Jan. 2010, pp. 368–372.
[29] W. S. Lee, P. L. Bartlett, and R. C. Williamson, "Efficient agnostic learning of neural networks with bounded fan-in," *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 2118–2132, Nov. 1996.
[30] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
[31] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput.*, Nov. 1993, pp. 40–44.
[32] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.
[33] G. Reeves and M. Gastpar, "Approximate sparsity pattern recovery: Information-theoretic lower bounds," *IEEE Trans. Inf. Theory*, vol. 59, no. 6, pp. 3451–3465, Jun. 2013.
[34] G. Reeves and M. Gastpar, "Fundamental tradeoffs for sparsity pattern recovery," *Inf. Transf. Manag.*, Jun. 2010.
[35] G. Reeves and M. Gastpar, "The sampling rate-distortion tradeoff for sparsity pattern recovery in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 3065–3092, May 2012.
[36] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 5, no. 1, pp. 3–55, 2001.
[37] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Statist. Soc. Ser. B, Statist. Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
[38] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.
[39] J. A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1030–1051, Mar. 2006.
[40] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.
[41] R. Venkataramanan, A. Joseph, and S. Tatikonda, "Gaussian rate-distortion via sparse linear regression over compact dictionaries," in *Proc. IEEE ISIT*, Jul. 2012, pp. 368–372.
[42] R. Venkataramanan, T. Sarkar, and S. Tatikonda, "Lossy compression via sparse linear regression: Computationally efficient encoding and decoding," Dec. 2012.
[43] M. J. Wainwright, "Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting," *IEEE Trans. Inf. Theory*, vol. 55, no. 12, pp. 5728–5741, Dec. 2009.
[44] M. J. Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (Lasso)," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2183–2202, May 2009.
[45] C. H. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *Ann. Statist.*, vol. 38, no. 2, pp. 894–942, 2010.
[46] T. Zhang, *Adaptive Forward-Backward Greedy Algorithm for Sparse Learning with Linear Models*. Lake Tahoe, NV, USA: NIPS, 2008.
[47] P. Zhao and B. Yu, "On model selection consistency of lasso," *J. Mach. Learn. Res.*, vol. 7, pp. 2541–2563, Nov. 2006.

Antony Joseph, biography not available at the time of publication.

Andrew R. Barron, biography not available at the time of publication.