Rateless Coding for Gaussian Channels

Uri Erez, *Member, IEEE*, Mitchell D. Trott, *Fellow, IEEE*, and Gregory W. Wornell, *Fellow, IEEE*

***Abstract*—A rateless code—i.e., a rate-compatible family of codes—has the property that codewords of the higher rate codes are prefixes of those of the lower rate ones. A perfect family of such codes is one in which each of the codes in the family is capacity-achieving. We show by construction that perfect rateless codes with low-complexity decoding algorithms exist for additive white Gaussian noise channels. Our construction involves the use of layered encoding and successive decoding, together with repetition using time-varying layer weights. As an illustration of our framework, we design a practical three-rate code family. We further construct rich sets of near-perfect rateless codes within our architecture that require either significantly fewer layers or lower complexity than their perfect counterparts. Variations of the basic construction are also developed, including one for time-varying channels in which there is no *a priori* stochastic model.**

***Index Terms*—Incremental redundancy, rate-compatible punctured codes, hybrid ARQ (H-ARQ), static broadcasting.**

I. Introduction

THE design of effective "rateless" codes has received renewed strong interest in the coding community, motivated by a number of emerging applications. Such codes have a long history, and have gone by various names over time, among them incremental redundancy codes, rate-compatible punctured codes, hybrid automatic repeat request (ARQ) type II codes, and static broadcast codes [1]–[10]. This paper focuses on the design of such codes for average power limited additive white Gaussian noise (AWGN) channels. Specifically, we develop techniques for mapping standard good single-rate codes for the AWGN channel into good rateless codes that are efficient, practical, and can operate at rates of multiple b/s/Hz. As such, they represent an attractive alternative to traditional hybrid ARQ solutions for a variety of wireless and related applications.

More specifically, we show that the successful techniques employed to construct low-complexity codes for the standard AWGN channel—such as those arising out of turbo and low-density parity check (LDPC) codes—can be leveraged to construct rateless codes. In particular, we develop an architecture in which a single codebook designed to operate at a single SNR is used in a straightforward manner to build a rateless codebook that operates at many SNRs.

The encoding in our architecture exploits three key ingredients: layering, repetition, and time-varying weighting. By layering, we mean the creation of a code by a linear combination of subcodes. By repetition, we mean the use of simple linear redundancy. Finally, by time-varying weighting, we mean that the (complex) weights in the linear combinations in each copy are different. We show that with the appropriate combination of these ingredients, if the base codes are capacity-achieving, so will be the resulting rateless code.

In addition to achieving capacity in our architecture, we seek to ensure that if the base code can be decoded with low complexity, so can the rateless code. This is accomplished by imposing the constraint that the layered encoding be successively decodable—i.e., that the layers can be decoded one at a time, treating as yet undecoded layers as noise.

Hence, our main result is the construction of capacity-achieving, low-complexity rateless codes, i.e., rateless codes constructed from layering, repetition, and time-varying weighting, that are successively decodable.

The paper is organized as follows. In Section II we put the problem in context and summarize related work and approaches. In Section III we introduce the channel and system model. In Section IV we motivate and illustrate our construction with a simple special-case example. In Section V we develop our general construction and show that within it exist perfect rateless codes for at least some ranges of interest, and in Section VI we develop and analyze specific instances of our codes generated numerically. In Section VII, we show that within the constraints of our construction rateless codes for any target ceiling rate and range can be constructed that are arbitrarily close to perfect in an appropriate sense. In Section VIII we make some comments on design and implementation issues, and in Section IX we describe the results of simulations with our constructions. In Section X, we discuss and develop simple extensions of our basic construction to time-varying channels. Finally, Section XI provides some concluding remarks.

II. Background

From a purely information theoretic perspective the problem of rateless transmission is well understood; see Shulman [11] for a comprehensive treatment. Indeed, for channels having one maximizing input distribution, a codebook drawn independently and identically distributed (i.i.d.) at random from this distribution will be good with high probability, when truncated to (a finite number of) different lengths. Phrased differently, in such cases random codes are rateless codes.

Constructing good codes that also have computationally efficient encoders and decoders requires more effort. A remark-

Manuscript received August 20, 2007; revised December 20, 2010; accepted September 16, 2011. Date of current version February 08, 2012. This work was supported in part by the National Science Foundation under Grant CCF-0515122, Draper Laboratory, MITRE Corp., and by Hewlett-Packard Co. through the MIT/HP Alliance. This work was presented in part at the Information Theory and Applications Workshop, University of California, San Diego, Feb. 2006, and at the 2006 IEEE International Symposium on Information Theory.

U. Erez is with the Department of Electrical Engineering—Systems, Tel Aviv University, Ramat Aviv, 69978, Israel (e-mail: uri@eng.tau.ac.il).

M. D. Trott is with Hewlett-Packard Laboratories, Palo Alto, CA, 94304 USA (e-mail: mitchell.trott@hp.com).

G. W. Wornell is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: gww@mit.edu).

Communicated by T. J. Richardson, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2011.2173242

able example of such codes for *erasure* channels are the recent Raptor codes of Shokrollahi [12], which build on the LT codes of Luby [13], [14]. An erasure channel model (for packets) is most appropriate for rateless coding architectures anchored at the application layer, where there is little or no access to the physical layer.

Apart from erasure channels, there is a growing interest in exploiting rateless codes closer to the physical layer, where AWGN models are more natural; see, e.g., [15] and the references therein. Much less is known about the limits of what is possible in this realm, which has been the focus of traditional hybrid ARQ research.

One line of work involves extending Raptor code constructions to binary-input AWGN channels (among others). In this area, [16], [17] have shown that no degree distribution allows such codes to approach capacity simultaneously at different signal to noise ratios (SNRs). Nevertheless, this does not rule out the possibility that such codes, when suitably designed, can be near capacity at multiple SNRs.

A second approach is based on puncturing of low-rate capacity-approaching binary codes such as turbo and LDPC codes [3], [8], [9], [15], [18], [19], or extending a higher-rate such code, or using a combination of code puncturing and extension [20]. When iterative decoding is involved, such approaches lead to performance tradeoffs at different rates—improving performance at one rate comes at the expense of the performance at other rates. While practical codes have been constructed in this manner [3], [20], it remains to be understood how close, in principle, one can come to capacity simultaneously at multiple SNRs, particularly when not all SNRs are low.

Finally, for the higher rates typically of interest, which necessitate higher-order (e.g., 16-QAM and larger) constellations, the modulation used with such binary codes becomes important. In turn, such modulation tends to further complicate the iterative decoding, imposing additional code design challenges. Constellation rearrangement and other techniques have been developed to at least partially address such challenges [21]–[24], but as yet do not offer a complete solution. Alternatively, suitably designed binary codes can, in principle, be combined with bit-interleaved coded modulation (BICM) for such applications; for example, [25] explores the design of Raptor codes for this purpose, and shows by example that the gaps to capacity need not be too large, at least provided the rates are not too high.

From the perspective of the broader body of related work described above, the present paper represents somewhat of a departure in approach to the design of rateless codes and hybrid ARQ systems. However, with this departure come additional complementary insights, as we will develop.

III. Channel and System Model

The codes we construct are designed for a complex AWGN channel

$$\mathbf{y}_m = \beta \mathbf{x}_m + \mathbf{z}_m, \qquad m = 1, 2, \ldots \tag{1}$$

where β is a channel gain,[1] $\mathbf{x}_m$ is a vector of N input symbols, $\mathbf{y}_m$ is the vector of channel output symbols, and $\mathbf{z}_m$ is a noise vector of N i.i.d. complex, circularly-symmetric Gaussian random variables of variance σ^2, independent across blocks $m = 1, 2, \ldots$. The channel input is limited to average power P per symbol. In our model, the channel gain β and noise variance σ^2 are known *a priori* at the receiver but not at the transmitter.[2]

The block length N has no important role in the analysis that follows. It is, however, the block length of the base code used in the rateless construction. As the base code performance controls the overall code performance, to approach channel capacity N must be large.

The encoder transmits a message w by generating a sequence of code blocks (incremental redundancy blocks) $\mathbf{x}_1(w)$, $\mathbf{x}_2(w), \ldots$. The receiver accumulates sufficiently many received blocks $\mathbf{y}_1, \mathbf{y}_2, \ldots$ to recover w. The channel gain β may be viewed as a variable parameter in the model; more incremental redundancy is needed to recover w when β is small than when β is large.

An important feature of this model is that the receiver always starts receiving blocks from index $m = 1$. It does not receive an arbitrary subsequence of blocks, as might be the case if one were modeling a broadcast channel that permits "tuning in" to an ongoing transmission.

We now define some basic terminology and notation. Unless noted otherwise, all logarithms are base 2, all symbols denote complex quantities, and all rates are in bits per complex symbol (channel use), i.e., b/s/Hz. We use $\cdot^{\mathrm{T}}$ for transpose and $\cdot^{\dagger}$ for Hermitian (conjugate transpose) operators. Vectors and matrices are denoted using bold face, random variables are denoted using sans-serif fonts, while sample values use regular (serif) fonts.

We define the *ceiling rate* of the rateless code as the highest rate R at which the code can operate, i.e., the effective rate if the message is decoded from the single received block y_1; hence, a message consists of NR information bits. Associated with this rate is an SNR *threshold*, which is the minimum SNR required in the realized channel for decoding to be possible from this single block. This SNR threshold can equivalently be expressed in the form of a channel gain threshold. Similarly, if the message is decoded from $m \geq 2$ received blocks, the corresponding effective code rate is R/m, and there is a corresponding SNR (and channel gain) threshold. Thus, for a rateless encoding consisting of M blocks, there is a sequence of M associated SNR thresholds.

Finally, as in the introduction, we refer to the code out of which our rateless construction is built as the *base code*, and the associated rate of this code as simply the *base code rate*. At points in our analysis we assume that a good base code is used in the code design, i.e., that the base code is capacity-achieving for the AWGN channel, and thus has the associated properties of such codes. This allows us to distinguish losses due to the code architecture from those due to the choice of base code.

[1] More general models for β will be discussed later in the paper.

[2] An equivalent model would be a broadcast channel in which a single encoding of a common message is being sent to a multiplicity of receivers, each experiencing a different SNR.

IV. Motivating Example

To develop initial insights, we construct a simple low-complexity perfect rateless code that employs two layers of coding to support a total of two redundancy blocks.

We begin by noting that for the case of a rateless code with two redundancy blocks the channel gain $|\beta|$ may be classified into three intervals based on the number of blocks needed for decoding. Let α_1 and α_2 denote the two associated channel gain thresholds. When $|\beta| \geq \alpha_1$ decoding requires only one block. When $\alpha_1 > |\beta| \geq \alpha_2$ decoding requires two blocks. When $\alpha_2 > |\beta|$ decoding is not possible. The interesting cases occur when the gain is as small as possible to permit decoding. At these threshold values, for one-block decoding the decoder sees (aside from an unimportant phase shift)

$$\mathbf{y}_1 = \alpha_1 \mathbf{x}_1 + \mathbf{z}_1 \tag{2}$$

while for two-block decoding the decoder sees

$$\mathbf{y}_1 = \alpha_2 \mathbf{x}_1 + \mathbf{z}_1 \tag{3}$$
$$\mathbf{y}_2 = \alpha_2 \mathbf{x}_2 + \mathbf{z}_2. \tag{4}$$

In general, given any particular choice of the ceiling rate R for the code, we would like the resulting SNR thresholds to be as low as possible. To determine lower bounds on these thresholds, let

$$\mathrm{SNR}_m = P\alpha_m^2/\sigma^2 \tag{5}$$

and note that the capacity of the one-block channel is

$$I_1 = \log(1 + \mathrm{SNR}_1) \tag{6}$$

while for the two-block channel the capacity is

$$I_2 = 2\log(1 + \mathrm{SNR}_2) \tag{7}$$

bits per channel use. A "channel use" in the second case consists of a pair of transmitted symbols, one from each block.

In turn, since we deliver the same message to the receiver for both the one- and two-block cases, the smallest values of α_1 and α_2 we can hope to achieve occur when

$$I_1 = I_2 = R. \tag{8}$$

Thus, we say that the code is *perfect* if it is decodable at these limits.

We next impose that the construction be a *layered* code, and that the layers be *successively decodable*.

Layering means that we require the transmitted blocks to be linear combinations of two base codewords $\mathbf{c}_1 \in \mathcal{C}_1$ and $\mathbf{c}_2 \in \mathcal{C}_2$, as follows:[3]

$$\mathbf{x}_1 = g_{11}\mathbf{c}_1 + g_{12}\mathbf{c}_2 \tag{9}$$
$$\mathbf{x}_2 = g_{21}\mathbf{c}_1 + g_{22}\mathbf{c}_2. \tag{10}$$

[3]In practice, the codebooks $\mathcal{C}_1$ and $\mathcal{C}_2$ should not be identical, though they can for example be derived from a common base codebook via scrambling. This point is discussed further in Section VIII.

Base codebook $\mathcal{C}_1$ has rate R_1 and base codebook $\mathcal{C}_2$ has rate R_2, where $R_1 + R_2 = R$, so that total rate of the two codebooks equals the ceiling rate. We assume for this example that both codebooks are capacity-achieving, so that the codeword components are i.i.d. Gaussian. Furthermore, for convenience, we scale the codebooks to have unit power, so the power constraint instead enters through the constraints

$$|g_{11}|^2 + |g_{12}|^2 = P \tag{11}$$
$$|g_{21}|^2 + |g_{22}|^2 = P. \tag{12}$$

Finally, the successive decoding constraint in our system means that the layers are decoded one at a time to keep complexity low (on order of the base code complexity). Specifically, the decoder first recovers $\mathbf{c}_2$ while treating $\mathbf{c}_1$ as additive Gaussian noise, then recovers $\mathbf{c}_1$ using $\mathbf{c}_2$ as side information.

We now show that perfect rateless codes are possible within these constraints by constructing a matrix $\mathbf{G} = [g_{ml}]$ so that the resulting code satisfies (8). Finding an admissible $\mathbf{G}$ is simply a matter of some algebra: in the one-block case we need

$$R_1 = I_{\alpha_1}(c_1; y_1 \,|\, c_2) \tag{13}$$
$$R_2 = I_{\alpha_1}(c_2; y_1) \tag{14}$$

and in the two-block case we need

$$R_1 = I_{\alpha_2}(c_1; y_1, y_2 \,|\, c_2) \tag{15}$$
$$R_2 = I_{\alpha_2}(c_2; y_1, y_2). \tag{16}$$

The subscripts α_1 and α_2 are a reminder that these mutual information expressions depend on the channel gain, and the scalar variables denote individual components from the input and output vectors.

While evaluating (13)–(15) is straightforward, calculating the more complicated (16), which corresponds to decoding $\mathbf{c}_2$ in the two-block case, can be circumvented by a little additional insight. In particular, while $\mathbf{c}_1$ causes the effective noise in the two blocks to be correlated, observe that a capacity-achieving code requires x_1 and x_2 to be i.i.d. Gaussian. As c_1 and c_2 are Gaussian, independent, and equal in power by assumption, this occurs only if the rows of $\mathbf{G}$ are orthogonal. Moreover, the power constraint P ensures that these orthogonal rows have the same norm, which implies that $\mathbf{G}$ is a scaled unitary matrix.

The unitary constraint has an immediate important consequence: the per-layer rates R_1 and R_2 must be equal, i.e.,

$$R_1 = R_2 = R/2. \tag{17}$$

This occurs because the two-block case decomposes into two parallel orthogonal channels of equal SNR. We see in the next section that a comparable result holds for any number of layers.

From the definitions of SNR_1 and I_1 [cf. (5) and (6)], and the equality $I_1 = R$ (8), we find that

$$P\alpha_1^2/\sigma^2 = 2^R - 1. \tag{18}$$

Also, from (13) and (17), we find that

$$|g_{11}|^2\alpha_1^2/\sigma^2 = 2^{R/2} - 1. \tag{19}$$

Combining (18) and (19) yields

$$|g_{11}|^2 = P\frac{2^{R/2}-1}{2^R-1} = \frac{P}{2^{R/2}+1}. \tag{20}$$

The constraint that $\mathbf{G}$ be a scaled unitary matrix, together with the power constraint P, implies

$$|g_{12}|^2 = P - |g_{11}|^2 \tag{21}$$
$$|g_{21}|^2 = P - |g_{11}|^2 \tag{22}$$
$$|g_{22}|^2 = |g_{11}|^2 \tag{23}$$

which completely determines the squared modulus of the entries of $\mathbf{G}$.

Now, the mutual information expressions (13)–(16) are unaffected by applying a common complex phase shift to any row or column of $\mathbf{G}$, so without loss of generality we take the first row and first column of $\mathbf{G}$ to be real and positive. For $\mathbf{G}$ to be a scaled unitary matrix, g_{22} must then be real and negative. We have thus shown that, if a solution to (13)–(16) exists, it must have the form

$$\mathbf{G} = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \sqrt{\frac{P}{2^{R/2}+1}} \begin{bmatrix} 1 & 2^{R/4} \\ 2^{R/4} & -1 \end{bmatrix}. \tag{24}$$

Conversely, it is straightforward to verify that (13)–(16) are satisfied with this selection. Thus (24) characterizes the (essentially) unique solution $\mathbf{G}$.[4]

In summary, we have constructed a two-layer, two-block perfect rateless code from linear combinations of codewords drawn from equal-rate codebooks. Moreover, decoding can proceed one layer at a time with no loss in performance, provided the decoder is cognizant of the correlated noise caused by undecoded layers. In the sequel we consider the generalization of our construction to an arbitrary number of layers and redundancy blocks.

V. Rateless Codes With Layered Encoding and Successive Decoding

The rateless code construction we pursue is as follows [26]. First, we choose the range (maximum number M of redundancy blocks), the ceiling rate R, the number of layers L, and finally the associated codebooks $\mathcal{C}_1, \ldots, \mathcal{C}_L$. We will see presently that the L base codebooks must have equal rate R/L when constructing perfect rateless codes with $M = L$, and in any case using equal rates has the advantage of allowing the codebooks for each layer to be derived from a single base code.

Given codewords $\mathbf{c}_l \in \mathcal{C}_l, l = 1, \ldots, L$, the redundancy blocks $\mathbf{x}_1, \ldots, \mathbf{x}_M$ take the form

$$\begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_M \end{bmatrix} = \mathbf{G} \begin{bmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_L \end{bmatrix} \tag{25}$$

where $\mathbf{G}$ is an $M \times L$ matrix of complex gains and where $\mathbf{x}_m$ for each m and $\mathbf{c}_l$ for each l are row vectors of length N. The power constraint enters by limiting the rows of $\mathbf{G}$ to have squared norm P and by normalizing the codebooks to have unit power. With this notation, the elements of the mth row of $\mathbf{G}$ are the weights used in constructing the mth redundancy block from the L codewords.[5] In the sequel we use g_{ml} to denote the (m,l)th entry of $\mathbf{G}$ and $\mathbf{G}_{m,l}$ to denote the upper-left $m \times l$ submatrix of $\mathbf{G}$.[6]

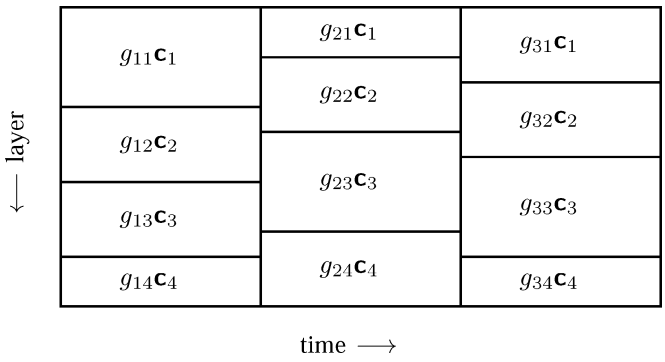

Fig. 1. Rateless code construction with four layers and three blocks of redundancy. Each block is a weighted linear combination of the (N-element) base codewords $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_4$, where g_{ml}, the (m,l)th element of $\mathbf{G}$, denotes the weight for layer l of block m. In this illustration, the thickness of a layer is a graphical depiction of the magnitude of its associated gain (power).

An example of this layered rateless code structure is depicted in Fig. 1. Each redundancy block contains a repetition of the codewords used in the earlier blocks, but with a different complex scaling factor. The code structure may therefore be viewed as a hybrid of layering and repetition. Note that, absent assumptions on the decoder, the order of the layers is not important.

In addition to the layered code structure, there is additional decoding structure, namely that the layered code be successively decodable. Specifically, to recover the message, we first decode $\mathbf{c}_L$, treating $\mathbf{G}[\mathbf{c}_1^{\mathrm{T}} \cdots \mathbf{c}_{L-1}^{\mathrm{T}}]^{\mathrm{T}}$ as (colored) noise, then decode $\mathbf{c}_{L-1}$, treating $\mathbf{G}[\mathbf{c}_1^{\mathrm{T}} \cdots \mathbf{c}_{L-2}^{\mathrm{T}}]^{\mathrm{T}}$ as noise, and so on. Thus, our aim is to select $\mathbf{G}$ so that capacity is achieved for any number $m = 1, \ldots, M$ of redundancy blocks subject to the successive decoding constraint. Minimum mean-square error (MMSE) combining of the available redundancy blocks conveniently exploits the repetition structure in the code when decoding each layer.

Both the layered repetition structure (25) and the successive decoding constraint impact the degree to which we can approach a perfect code. Accordingly, we examine the consequences of each in turn.

We begin by examining the implications of the layered repetition structure (25). When the number of layers L is at least as large as the number of redundancy blocks M, such layering does not limit code performance. But when $L < M$, it does. In particular, whenever the number m of redundancy blocks required by the realized channel exceeds L, there is necessarily a gap between the code performance and capacity. To see this,

[4]Interestingly, the symmetry in (24) implies that the construction remains perfect even if the two redundancy blocks are received in swapped order. This is not true of our other constructions.

[5]The lth column of $\mathbf{G}$ also has a useful interpretation. In particular, one can interpret the construction as equivalent to a "virtual" code-division multiple-access (CDMA) system with L users, each corresponding to one layer of the rateless code. With this interpretation, the signature (spreading) sequence for the lth virtual user is the lth column of $\mathbf{G}$.

[6]Where necessary, we adopt the convention that $\mathbf{G}_{m,0} = \mathbf{0}$.

observe that (25) with (1), restricted to the first m blocks, defines a linear L-input m-output AWGN channel, the capacity of which is at most

$$I'_m = \begin{cases} m \log\left(1 + \frac{|\beta|^2 P}{\sigma^2}\right) & \text{for } m \le L \\ L \log\left(1 + \frac{m}{L}\frac{|\beta|^2 P}{\sigma^2}\right) & \text{for } m > L. \end{cases} \tag{26}$$

Only for $m \le L$ does this match the capacity of a general m-block AWGN channel, viz.,

$$I_m = m \log\left(1 + \frac{|\beta|^2 P}{\sigma^2}\right). \tag{27}$$

Ultimately, for $m > L$ the problem is that an L-fold linear combination cannot fill all degrees of freedom afforded by the m-block channel.

An additional penalty occurs when we combine the layered repetition structure with the requirement that the code be rateless. Specifically, for $M > L$, there is no choice of gain matrix $\mathbf{G}$ that permits (26) to be met with equality *simultaneously* for all $m = 1, \ldots, M$. A necessary and sufficient condition for equality is that the rows of $\mathbf{G}_{m,L}$ be orthogonal for $m \le L$ and the columns of $\mathbf{G}_{m,L}$ be orthogonal for $m > L$. This follows because reaching (26) for $m \le L$ requires that the linear combination of L codebooks create an i.i.d. Gaussian sequence. In contrast, reaching (26) for $m > L$ requires that the linear combination inject the L codebooks into orthogonal subspaces, so that a fraction L/m of the available degrees of freedom are occupied by i.i.d. Gaussians (the rest being empty).

Unfortunately, the columns of $\mathbf{G}_{m,L}$ cannot be orthogonal simultaneously for all $m > L$; orthogonal m-dimensional vectors (with nonzero entries) cannot remain orthogonal when truncated to their first $m-1$ dimensions. Thus (26) determines only a lower bound on the loss due to the layering structure (25). Fortunately, the additional loss encountered in practice turns out to be quite small, as we demonstrate numerically as part of the next section.

When $M = L$, the orthogonality requirement forces $\mathbf{G}$ to be a scaled unitary matrix. Upon receiving the final redundancy block $m = M$, the problem decomposes into L parallel channels with equal SNR, which in turn implies that the rate of each layer must equal R/L.

A lower bound on loss incurred by the use of insufficiently many layers is readily obtained by comparing (26) and (27). Given a choice of ceiling rate R for the rateless code, (26) implies that for rateless codes constructed using linear combinations of L base codes, the smallest channel gain α'_m for which it's possible to decode with m blocks is

$$\alpha'^2_m = \begin{cases} (2^{R/m} - 1)\frac{\sigma^2}{P} & \text{for } m \le L \\ (2^{R/L} - 1)\frac{L}{m}\frac{\sigma^2}{P} & \text{for } m > L. \end{cases} \tag{28}$$

By comparison, (27) implies that without the layering constraint the corresponding channel gain thresholds α_m are

$$\alpha^2_m = (2^{R/m} - 1)\frac{\sigma^2}{P}. \tag{29}$$

The resulting performance loss α'_m/α_m caused by the layered structure as calculated from (28) and (29) is shown in decibels in Table I for a target ceiling rate of $R = 5$ bits/symbol. For example, if an application requires $M = 10$ redundancy blocks, a 3-layer code has a loss of less than 2 dB at $m = 10$, while a 5-layer code has a loss of less than 0.82 dB at $m = 10$.

TABLE I
LOSSES α'_m/α_m IN DECIBELS DUE TO LAYERED STRUCTURE IMPOSED ON A RATELESS CODE OF CEILING RATE $R = 5$ B/S/HZ, AS A FUNCTION OF THE NUMBER OF LAYERS L AND REDUNDANCY BLOCKS m

	Redundancy blocks m								
	2	3	4	5	6	7	8	9	10
$L=1$	5.22	6.77	7.50	7.92	8.20	8.40	8.54	8.65	8.74
$L=2$	0.00	1.55	2.28	2.70	2.98	3.17	3.32	3.43	3.52
$L=3$	0.00	0.00	0.73	1.16	1.43	1.63	1.77	1.88	1.97
$L=4$	0.00	0.00	0.00	0.42	0.70	0.90	1.04	1.15	1.24
$L=5$	0.00	0.00	0.00	0.00	0.28	0.47	0.62	0.73	0.82
$L=6$	0.00	0.00	0.00	0.00	0.00	0.20	0.34	0.45	0.54
$L=7$	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.26	0.35
$L=8$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.20
$L=9$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09

As Table I reflects—and as can be readily verified analytically—for a fixed number of layers L and a fixed base code rate R/L, the performance loss α'_m/α_m attributable to the imposition of layered encoding grows monotonically with the number of blocks m, approaching the limit

$$\frac{\alpha'^2_\infty}{\alpha^2_\infty} = \frac{2^{R/L} - 1}{(R/L)\ln 2}. \tag{30}$$

Thus, in applications where the number of incremental redundancy blocks is very large, it's advantageous to keep the base code rate small. For example, with a base code rate of 1/2 bit per complex symbol (implemented, for example, using a rate-1/4 binary code) the loss due to layering is at most 0.78 dB, while with a base code rate of 1 bit per complex symbol the loss is at most 1.6 dB.

We now determine the additional impact the successive decoding requirement has on our ability to approach capacity, and more generally what constraints it imposes on $\mathbf{G}$. We continue to incorporate the power constraint by taking the rate-R/L codebooks $\mathcal{C}_1, \ldots, \mathcal{C}_L$ to have unit power and the rows of $\mathbf{G}$ to have squared norm P. Since our aim is to employ codebooks designed for (non-fading) Gaussian channels, we make the further assumption that the codebooks have constant power, i.e., that they satisfy the per-symbol energy constraint $E[|c_{l,n}(\mathsf{w})|^2] \le 1$ for all layers l and time indices $n = 1, \ldots, N$, where the expectation is taken over equiprobable messages $\mathsf{w} \in \{1, \ldots, 2^{NR/L}\}$. Additional constraints on $\mathbf{G}$ now follow from the requirement that the mutual information accumulated through any block m at each layer l be large enough to permit successive decoding.

Concretely, suppose we have received blocks $1, \ldots, m$. Let the optimal threshold channel gain α_m be defined as in (29). Suppose further that layers $l+1, \ldots, L$ have been successfully decoded, and define

$$\begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_m \end{bmatrix} = \beta \mathbf{G}_{m,l} \begin{bmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_l \end{bmatrix} + \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_m \end{bmatrix} \tag{31}$$

as the received vectors without the contribution from layers $l+1, \ldots, L$.

Then, following standard arguments, with independent equiprobable messages for each layer, the probability of decoding error for layers $1,\dots,l$ can be made vanishingly small with increasing block length only if the mutual information between input and output is at least as large as the combined rate lR/L of the codes $\mathcal{C}_1,\dots,\mathcal{C}_l$. That is, when β equals the optimal threshold gain α_m, successive decoding requires

$$lR/L \le (1/N)I\left(\mathbf{c}_1,\dots,\mathbf{c}_l;\mathbf{y}_1,\dots,\mathbf{y}_m \,\middle|\, \mathbf{c}_{l+1}^L\right) \tag{32}$$

$$= (1/N)I(\mathbf{c}_1,\dots,\mathbf{c}_l;\mathbf{v}_1,\dots,\mathbf{v}_m) \tag{33}$$

$$= (1/N)(H(\mathbf{v}_1,\dots,\mathbf{v}_m) - H(\mathbf{v}_1,\dots,\mathbf{v}_m \,|\, \mathbf{c}_1,\dots,\mathbf{c}_l)) \tag{34}$$

$$\le \log\det\left(\sigma^2\mathbf{I} + \alpha_m^2\mathbf{G}_{m,l}\mathbf{G}_{m,l}^\dagger\right) - \log\det(\sigma^2\mathbf{I}) \tag{35}$$

$$= \log\det\left(\mathbf{I} + \left(\alpha_m^2/\sigma^2\right)\mathbf{G}_{m,l}\mathbf{G}_{m,l}^\dagger\right) \tag{36}$$

where $\mathbf{I}$ is an appropriately sized $(m \times m)$ identity matrix. The inequality (35) relies on the assumption that the codebooks have constant power, and it holds with equality if the components of $\mathbf{G}_{m,l}[\mathbf{c}_1^T,\dots,\mathbf{c}_l^T]^T$ are jointly Gaussian, which by Cramer's theorem requires the components of $\mathbf{c}_1,\dots,\mathbf{c}_l$ to be jointly Gaussian.

Our ability to choose $\mathbf{G}$ to either exactly or approximately satisfy (36) for all $l = 1,\dots,L$ and each $m = 1,\dots,M$ determines the degree to which we can approach capacity. It is straightforward to see that there is no slack in the problem; (36) can be satisfied simultaneously for all l and m only if the inequalities are all met with equality. Beyond this observation, however, the conditions under which (36) may be satisfied are not obvious.

Characterizing the set of solutions for $\mathbf{G}$ when $L = M = 2$ was done in Section IV [see (24)]. Characterizing the set of solutions when $L = M = 3$ requires more work. It is shown in Appendix A that, when it exists, a solution $\mathbf{G}$ must have the form

$$\mathbf{G} = \sqrt{x-1} \cdot \begin{bmatrix} \sqrt{x+1} & \sqrt{x^2(x+1)} & \sqrt{x^4(x+1)} \\ \sqrt{x^3(x+1)} & e^{j\theta_1}\sqrt{x^5+1} & e^{j\theta_2}\sqrt{x(x+1)} \\ \sqrt{x^2(x^3+1)} & e^{j\theta_3}\sqrt{x(x^3+1)} & e^{j\theta_4}\sqrt{x^3+1} \end{bmatrix} \tag{37}$$

where $x = 2^{R/6}$ and where $e^{j\theta_i}, i = 1,\dots,4$ are complex phasors. The desired phasors—or a proof of nonexistence—may be determined from the requirement that $\mathbf{G}$ be a scaled unitary matrix. Using this observation, it is shown in Appendix A that a solution $\mathbf{G}$ exists and is unique (up to complex conjugate) for all $R \le 3(\log(7+3\sqrt{5})-1) \approx 8.33$ bits per complex symbol, but no choice of phasors results in a unitary $\mathbf{G}$ for larger values of R.

For example, using (37) with $R = 6$ bits/symbol we find that

$$P = 63, \quad \alpha_1 = 1, \quad \alpha_2 = \sqrt{1/9}, \quad \alpha_3 = \sqrt{1/21}$$

$$\mathbf{G} = \begin{bmatrix} \sqrt{3} & \sqrt{12} & \sqrt{48} \\ \sqrt{24} & \sqrt{33}e^{j\theta_1} & \sqrt{6}e^{j\theta_2} \\ \sqrt{36} & \sqrt{18}e^{j\theta_3} & \sqrt{9}e^{j\theta_4} \end{bmatrix}$$

where

$$\theta_1 = \arccos\frac{-5}{2\sqrt{22}} \qquad \theta_2 = 2\pi - \arctan 3\sqrt{7}$$

$$\theta_3 = -\arctan\sqrt{7} \qquad \theta_4 = \pi - \arctan\sqrt{7}/3.$$

TABLE II
PERCENT SHORTFALL IN RATE FOR A NUMERICALLY-OPTIMIZED RATELESS CODE WITH $M = 10$ BLOCKS, $L = 3$ LAYERS, AND A CEILING RATE OF $R = 5$ B/S/HZ

	Redundancy blocks m									
	1	2	3	4	5	6	7	8	9	10
$l=1$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
$l=2$	0.00	0.28	1.23	1.46	1.39	0.44	0.59	0.48	0.16	0.23
$l=3$	0.00	0.29	1.23	1.48	1.40	0.43	0.54	0.51	0.15	0.23

For $M > 3$ the algebra becomes daunting, though we conjecture that exact solutions and hence perfect rateless codes exist for all $L = M$, for at least some nontrivial values of R.[7]

For $L < M$ perfect constructions cannot exist. As developed earlier in this section, even if we replace the optimum threshold channel gains α_m defined via (29) with suboptimal gains α'_m of (28) determined by the layering bound (26), it is still not possible to satisfy (36). However, one can come close. While the associated analysis is nontrivial, such behavior is easily demonstrated numerically, which we show as part of the next section.

VI. NUMERICAL EXAMPLES

In this section, we consider numerical constructions both for the case $L = M$ and for the case $L < M$. Specifically, we have experimented with numerical optimization methods to satisfy (36) for up to $M = 10$ redundancy blocks, using the threshold channel gains α'_m defined via (28) in place of those defined via (29) as appropriate when the number of blocks M exceeds the number of layers L.

For the case $L = M$, for each of $M = 2,3,\dots,10$, we found constructions with $R/L = 2$ bits/symbol that come within 0.1% of satisfying (36) subject to (29), and often the solutions come within 0.01%. This provides powerful evidence that perfect rateless codes exist for a wide range of parameter choices.

For the case $L < M$, despite the fact that there do not exist perfect codes, in most cases of interest one can come remarkably close to satisfying (36) subject to (28). Evidently mutual information for Gaussian channels is quite insensitive to modest deviations of the noise covariance away from a scaled identity matrix.

As an example, Table II shows the rate shortfall in meeting the mutual information constraints (36) for an $L = 3$ layer code with $M = 10$ redundancy blocks, and a target ceiling rate $R = 5$. The associated complex gain matrix is

$$\mathbf{G} = \begin{bmatrix} 1.4747 & 2.6277 & 4.6819 \\ 3.5075 & 3.7794\,e^{j2.0510} & 2.1009\,e^{-j1.9486} \\ 4.0648 & 3.1298\,e^{-j0.9531} & 2.1637\,e^{j2.5732} \\ 3.2146 & 3.1322\,e^{j3.0765} & 3.2949\,e^{j0.9132} \\ 3.2146 & 3.3328\,e^{-j1.6547} & 3.0918\,e^{-j1.4248} \\ 3.2146 & 3.1049\,e^{j0.9409} & 3.3206\,e^{j2.8982} \\ 3.2146 & 3.3248\,e^{j1.2506} & 3.1004\,e^{-j0.2027} \\ 3.2146 & 3.0980\,e^{-j1.4196} & 3.3270\,e^{j1.9403} \\ 3.2146 & 3.2880\,e^{-j2.9449} & 3.1394\,e^{-j1.9243} \\ 3.2146 & 3.1795\,e^{j0.7839} & 3.2492\,e^{j0.3413} \end{bmatrix}.$$

The worst case loss is less than 1.5%; this example is typical in its efficiency.

[7]In recent calculations following the above approach, Ayal Hitron at Tel Aviv University has determined that exact solutions exist in the $M = L = 4$ case for rates in the range $R \le 10.549757$.

The total loss of the designed code relative to a perfect rateless code is, of course, the sum of the successive decoding and layered encoding constraint losses. Hence, the losses in Tables I and II are cumulative. As a practical matter, however, when $L < M$, the layered encoding constraint loss dwarfs that due to the successive decoding constraint: the overall performance loss arises almost entirely from the code's inability to occupy all available degrees of freedom in the channel. Thus, this overall loss can be estimated quite closely by comparing (27) and (26). Indeed this is reflected in our example, where the loss of Table I dominates over that of Table II.

VII. Existence of Near-Perfect Rateless Codes

While the closed-form construction of *perfect* rateless codes subject to layered encoding and successive decoding becomes more challenging with increasing code range M, the construction of codes that are at least nearly perfect is comparatively straightforward. In the preceding section, we demonstrated this numerically. In this section, we prove this analytically. In particular, we construct rateless codes (of any desired ceiling rate) that are arbitrarily close to perfect in an appropriate sense, provided enough layers are used. We term these near-perfect rateless codes. The code construction we present is applicable to arbitrarily large M and also allows for simpler decoding than that required in the preceding development.

The near-perfect codes we develop in this section [27] are closely related to those in Section V. However, there are a few differences. We retain the layered construction, but instead of using a single complex weight for the codeword at each layer (and block), we use a single weight magnitude for each codeword and vary the phase of the weight from symbol to symbol within the codeword in each layer (and block). Moreover, in our analysis, the phases are chosen randomly, corresponding to evaluating an ensemble of codes. The realizations of these random phases are known to and exploited by the associated decoders. As with the usual random coding development, we establish the existence of good codes in the ensemble by showing that the average performance is good.

These modifications, and in particular the additional degrees of freedom in the code design, simplify the analysis—at the expense of some slightly more cumbersome notation. Additionally, because of these differences, the particular gain matrices in this section cannot be easily compared with those of Section V, but we do not require such comparisons.

A. Encoding

As discussed above, in our approach to perfect constructions in Section V, we made each redundancy block a linear combination of the base codewords, where the weights are the corresponding row of the combining matrix $\mathbf{G}$, as (25) indicates. Each individual symbol of a particular redundancy block is, therefore, a linear combination of the corresponding symbols in the respective base codewords, with the combining matrix being the same for all such symbols.

Since for the codes of this section we allow the combining matrix to vary from symbol to symbol in the construction of each redundancy block, we augment our notation. In particular, using $\mathsf{c}_l(n)$ and $\mathsf{x}_m(n)$ to denote the nth elements of codeword $\mathbf{c}_l$ and redundancy block $\mathbf{x}_m$, respectively, we have [cf. (25)]

$$\begin{bmatrix} \mathsf{x}_1(n) \\ \vdots \\ \mathsf{x}_M(n) \end{bmatrix} = \mathbf{G}(n) \begin{bmatrix} \mathsf{c}_1(n) \\ \vdots \\ \mathsf{c}_L(n) \end{bmatrix}, \qquad n = 1, 2, \ldots, N. \quad (38)$$

The value of M plays no role in our development and may be taken arbitrarily large. Moreover, as before, the power constraint enters by limiting the rows of $\mathbf{G}(n)$ to have a squared norm P and by normalizing the codebooks to have unit power.

It suffices to restrict our attention to $\mathbf{G}(n)$ of the form

$$\mathbf{G}(n) = \mathbf{P} \odot \mathbf{D}(n) \quad (39)$$

where $\mathbf{P}$ is an $M \times L$ (deterministic) power allocation matrix with entries $\sqrt{p_{m,l}}$ that do not vary within a block

$$\mathbf{P} = \begin{bmatrix} \sqrt{p_{1,1}} & \cdots & \sqrt{p_{1,L}} \\ \vdots & \ddots & \vdots \\ \sqrt{p_{M,1}} & \cdots & \sqrt{p_{M,L}} \end{bmatrix} \quad (40)$$

and $\mathbf{D}(n)$ is a (random) phase-only "dither" matrix of the form

$$\mathsf{D}(n) = \begin{bmatrix} \mathsf{d}_{1,1}(n) & \cdots & \mathsf{d}_{1,L}(n) \\ \vdots & \ddots & \vdots \\ \mathsf{d}_{M,1}(n) & \cdots & \mathsf{d}_{M,L}(n) \end{bmatrix} \quad (41)$$

with $\odot$ denoting elementwise multiplication. In our analysis, the $\mathsf{d}_{ij}(n)$ are all i.i.d. in i, j, and n, and are independent of all other random variables, including noises, messages, and codebooks. As we shall see below, the role of the dither is to decorrelate pairs of random variables, hence it suffices for $\mathsf{d}_{ij}(n)$ to take values $+1$ and -1 with equal probability.

B. Decoding

To obtain a near-perfect rateless code, it is sufficient to employ a successive cancellation decoder with simple maximal ratio combining (MRC) of the redundancy blocks. While, in principle, an MMSE-based successive cancellation decoder enables higher performance, as we will see, an MRC-based one is sufficient for our purposes, and simplifies the analysis. Indeed, although the encoding we choose creates a per-layer channel that is time-varying, the MRC-based successive cancellation decoder effectively transforms the channel back into a time-invariant one, for which any of the traditional low-complexity capacity-approaching codes for the AWGN channel are suitable as a base code in the design.[8]

The decoder operation is as follows, assuming the SNR is such that decoding is possible from m redundancy blocks. To decode the Lth (top) layer, the dithering is first removed from the received waveform by multiplying by the conjugate dither

[8]More generally, the MRC-based decoder is particularly attractive for practical implementation. Indeed, as each redundancy block arrives a sufficient statistic for decoding can be accumulated without the need to retain earlier blocks in buffers. The computational cost of decoding thus grows linearly with block length while the memory requirements do not grow at all. This is much less complex than the MMSE decoder discussed in the development of the codes of Section V.

sequence for that layer. Then, the m blocks are combined into a single block via the appropriate MRC for that layer. The message in this Lth layer is then decoded, treating the undecoded layers as noise, and its contribution subtracted from the received waveform. The $(L-1)$st layer is now the top layer, and the process is repeated, until all layers have been decoded. Note that the use of MRC in decoding is equivalent to treating the undecoded layers as white (rather than structured) noise, which is the natural approach when the dither sequence structure in those undecoded (lower) layers is ignored in decoding the current layer of interest.

We now introduce notation that allows the operation of the decoder to be expressed more precisely. We then determine the effective SNR seen by the decoder at each layer of each redundancy block.

Since $\mathbf{G}(n)$ is drawn i.i.d., the overall channel is i.i.d., and thus we may express the channel model in terms of an arbitrary individual element in the block. Specifically, our received waveform can be expressed as [cf. (1) and (25)]

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} = \beta \mathbf{G} \begin{bmatrix} c_1 \\ \vdots \\ c_L \end{bmatrix} + \begin{bmatrix} z_1 \\ \vdots \\ z_M \end{bmatrix} \tag{42}$$

where $\mathbf{G} = \mathbf{P} \odot \mathbf{D}$, with $\mathbf{G}$ denoting the arbitrary element in the sequence $\mathbf{G}(n)$, and where y_m is the corresponding received symbol from redundancy block m (and similarly for $c_l, z_m, \mathbf{D}$).

If layers $l+1, l+2, \ldots, L$ have been successively decoded from m redundancy blocks, and their effects subtracted from the received waveform, the residual waveform is denoted by

$$\mathbf{v}_{m,l} = \beta \mathbf{G}_{m,l} \begin{bmatrix} c_1 \\ \vdots \\ c_l \end{bmatrix} + \begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix} \tag{43}$$

where we continue to let $\mathbf{G}_{m,l}$ denote the $m \times l$ upper-left submatrix of $\mathbf{G}$, and likewise for $\mathbf{D}_{m,l}$ and $\mathbf{P}_{m,l}$. As additional notation, we let $\mathbf{g}_{m,l}$ denote the m-vector formed from the upper m rows of the lth column of $\mathbf{G}$, whence

$$\mathbf{G}_{m,l} = [\, \mathbf{g}_{m,1} \quad \mathbf{g}_{m,2} \quad \cdots \quad \mathbf{g}_{m,l} \,] \tag{44}$$

and likewise for $\mathbf{d}_{m,l}$ and $\mathbf{p}_{m,l}$.

With such notation, the decoding can be expressed as follows. Starting with $\mathbf{v}_{m,L} = \mathbf{y}$, decoding proceeds. After layers $l+1$ and higher have been decoded and removed, we decode from $\mathbf{v}_{m,l}$. Writing

$$\mathbf{v}_{m,l} = \beta(\mathbf{d}_{m,l} \odot \mathbf{p}_{m,l}) c_l + \mathbf{v}_{m,l-1} \tag{45}$$

the operation of removing the dither can be expressed as

$$\mathbf{d}^*_{m,l} \odot \mathbf{v}_{m,l} = \beta \mathbf{p}_{m,l} c_l + \mathbf{v}'_{m,l-1} \tag{46}$$

where

$$\mathbf{v}'_{m,l-1} = \mathbf{d}^*_{m,l} \odot \mathbf{v}_{m,l-1}. \tag{47}$$

The MRC decoder treats the dither in the same manner as noise, i.e., as a random process with known statistics but unknown realization. Because the entries of the dither matrix are chosen to be i.i.d. random phases independent of the messages, the entries of $\mathbf{D}_{m,l}$ and $[\, c_1 \quad \cdots \quad c_{l-1} \,]$ are jointly and individually uncorrelated, and the effective noise $\mathbf{v}'_{m,l-1}$ seen by the MRC decoder has diagonal covariance $\mathbf{K}_{\mathbf{v}'_{m,l-1}} = E[\mathbf{v}'_{m,l-1} \mathbf{v}'^{\dagger}_{m,l-1}]$.

The effective SNR at which this lth layer is decoded from m blocks via MRC is thus

$$\mathrm{SNR}_{\mathrm{MRC}} = \sum_{m'=1}^{m} \mathrm{SNR}_{m',l}(\beta) \tag{48}$$

where

$$\mathrm{SNR}_{m',l}(\beta) = \frac{|\beta|^2 p_{m',l}}{|\beta|^2 (p_{m',1} + \ldots + p_{m',l-1}) + \sigma^2}. \tag{49}$$

Note that we have made explicit the dependency of these per-layer per-block SNRs on β.

C. Efficiency

The use of random dither at the encoder and MRC at the decoder both cause some loss in performance relative to the perfect rateless codes presented earlier. In this section we show that these losses can be made small.

When a coding scheme is not perfect, its *efficiency* quantifies how close the scheme is to perfect. There are ultimately several ways one could measure efficiency that are potentially useful for engineering design. Among these, we choose the following efficiency notion.

1) We find the ideal thresholds $\{\alpha_m\}$ for a perfect code of rate R.
2) We determine the highest rate R' such that an imperfect code designed at rate R' is decodable with m redundancy blocks when the channel gain is α_m, for all $m = 1, 2, \ldots$.
3) We measure efficiency η by the ratio R'/R, which is always less than unity.

With this notion of efficiency, we further define a coding scheme as near-perfect if the efficiency so-defined approaches unity when sufficiently many layers L are employed.

The efficiency of our scheme ultimately depends on the choice of our power allocation matrix (40). We now show the main result of this section: provided there exists a power allocation matrix such that for each l and m

$$\frac{R}{L} = \sum_{m'=1}^{m} \log(1 + \mathrm{SNR}_{m',l}(\alpha_m)) \tag{50}$$

with $\mathrm{SNR}_{m,l}(\cdot)$ as defined in (49), a near-perfect rateless coding scheme results. We prove the existence of such a power allocation—and develop an interpretation of (50)—in Appendix B, and thus focus on our main result in the sequel.

We establish our main result by finding a lower bound on the average mutual information between the input and output of the channel. Upon receiving m blocks with channel gain α_m, and assuming layers $l+1, \ldots, L$ are successfully decoded, let $I'_{m,l}$ be the mutual information between the input to the lth layer and the channel output. Then

$$I'_{l,m} = I(c_l; \mathbf{v}_{m,l} \,|\, \mathbf{d}_{m,l}) \tag{51}$$

$$= I(c_l; \alpha_m \mathbf{p}_{m,l} c_l + \mathbf{v}'_{m,l-1} \,|\, \mathbf{d}_{m,l}) \tag{52}$$

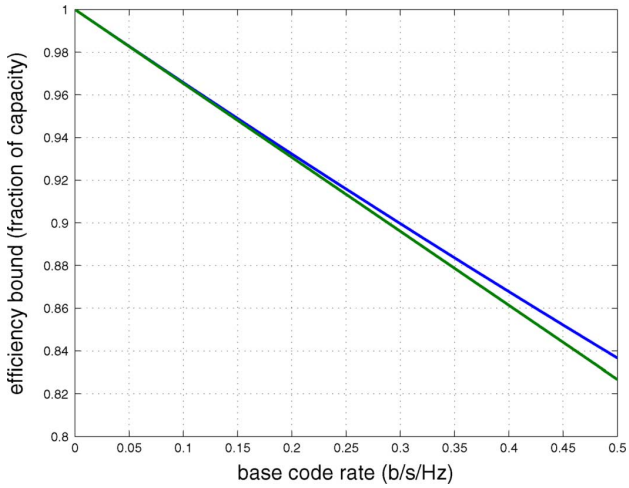

Fig. 2. Lower bound on efficiency of the near-perfect rateless code. The top and bottom curves are the middle and right-hand bounds of (59), respectively.

$$\geq I(c_l; \alpha_m \mathbf{p}_{m,l} c_l + \mathbf{v}'_{m,l}) \tag{53}$$
$$\geq I(c_l; \alpha_m \mathbf{p}_{m,l} c_l + \mathbf{v}''_{m,l}) \tag{54}$$
$$= \log(1 + \mathrm{SNR}_{\mathrm{MRC}}) \tag{55}$$

where (52) follows from (46)–(47), (53) follows from the independence of c_l and $\mathbf{d}_{m,l}$, and (54) obtains by replacing $\mathbf{v}'_{m,l-1}$ with a Gaussian random vector $\mathbf{v}''_{m,l-1}$ of covariance $\mathbf{K}_{\mathbf{v}'_{m,l-1}}$. Lastly, to obtain (55) we have used (48) for the post-MRC SNR.

Now, if the assumption (50) is satisfied, then the right-hand side of (55) is further bounded for all m by

$$I'_{m,l} \geq \log\left(1 + \ln 2 \frac{R}{L}\right) \tag{56}$$

where we have applied the inequality $\ln(1+u) \leq u$ (valid for $u > 0$) to (50) to conclude that $(\ln 2)R/L \leq \sum_{m'=1}^{m} \mathrm{SNR}_{m',l}(\alpha_m)$. Note that the lower bound (56) may be quite loose; for example, $I'_{m,l} = R/L$ when $m = 1$.

Thus, if we design each layer of the code for a base code rate of

$$\frac{R''}{L} = \log\left(1 + \ln 2 \frac{R}{L}\right) \tag{57}$$

(56) ensures decodability after m blocks are received when the channel gain is α_m, for $m = 1, 2, \ldots$.

Finally, rewriting (57) as

$$\frac{R}{L} = \frac{2^{R''/L} - 1}{\ln 2} \tag{58}$$

the efficiency η of the conservatively-designed layered repetition code is bounded by

$$\eta = \frac{R''}{R} = \frac{(\ln 2)R''/L}{2^{R''/L} - 1} \geq 1 - \frac{\ln 2}{2}\frac{R''}{L} \tag{59}$$

which approaches unity as $L \to \infty$ as claimed.

In Fig. 2, the efficiency bounds (59) are plotted as a function of the base code rate R''/L. As a practical matter, our bound implies, for instance, that to obtain 90% efficiency requires a base code of rate of roughly 1/3 bits per complex symbol. Note, too, that when the number of layers is sufficiently large that the SNR per layer is low, a binary code may be used instead of a Gaussian codebook, which may be convenient for implementation. For example, a code with rate 1/3 bits per complex symbol may be implemented using a rate-1/6 LDPC code with binary antipodal signaling.

VIII. Design and Implementation Issues

In this section, we comment on some issues that arise in the development and implementation of our rateless code constructions; additional implementation issues are addressed in [28].

First, one consequence of our development of perfect rateless codes for $M = L$ is that all layers must have the same rate R/L. This does not seem to be a serious limitation, as it allows a single base codebook to serve as the template for all layers, which in turn generally decreases the implementation complexity of the encoder and decoder. The codebooks $\mathcal{C}_1, \ldots, \mathcal{C}_L$ used for the L layers should not be identical, however, for otherwise a naive successive decoder might inadvertently swap messages from two layers or face other difficulties that increase the probability of decoding error. A simple cure to this problem is to apply pseudorandom phase scrambling to a single base codebook $\mathcal{C}$ to generate the different codebooks needed for each layer. Pseudorandom interleaving would have a similar effect.

Second, it should be emphasized that a layered code designed with the successive decoding constraint (36) can be decoded in a variety of ways. Because the undecoded layers act as colored noise, an optimal decoder should take this into account, for example by using a MMSE combiner on the received blocks $\{\mathbf{y}_m\}$ as mentioned in Section V. The MMSE combining weights change as each layer is stripped off. Alternatively, some or all of the layers could be decoded jointly; this might make sense when the decoder for the base codebook decoder is already iterative, and could potentially accelerate convergence compared to a decoder that treats the layers sequentially.

Third, a comparatively simple receiver is possible when all M blocks have been received from a perfect rateless code in which $M = L$. In this special case the linear combinations applied to the layers are orthogonal, hence the optimal receiver can decode each layer independently, without successive decoding. This property is advantageous in a multicasting scenario because it allows the introduction of users with simplified receivers that function only at certain rates, in this case the lowest supported one.

Finally, we note that with an ideal rateless code, *every* prefix of the code is a capacity-achieving code. This corresponds to a maximally dense set of SNR thresholds at which decoding can occur. By contrast, our focus in the paper has been on rateless codes that are capacity-achieving only for prefixes whose lengths are an integer multiple of the base block length. The associated sparseness of SNR thresholds can be undesirable in some applications, since when the realized SNR is between thresholds, there is no guarantee that capacity is achieved: the only realized rate promised by the construction is that corresponding to the next lower SNR threshold.

TABLE III
RATE 2/3 B/S/HZ 3GPP LTE BASE CODE EFFICIENCIES

	Bit-Error Rate ϵ			
	10^{-2}	10^{-3}	10^{-4}	10^{-5}
Efficiency $\eta_\circ$	88.9%	87.1%	85.7%	84.7%

However, as will be apparent from the simulations described in Section IX, performance is generally much better than this pessimistic assessment. In particular, partial blocks provide essentially all the necessary redundancy to allow an appropriately generalized decoder to operate as close to capacity as happens with full blocks.

Nevertheless, when precise control over the performance at a dense set of SNR thresholds is required, other approaches can be used. For example, when the target ceiling rate is R, we can use our rateless code construction to design a code of ceiling rate κR, where $1 \leq \kappa \leq M$, and have the decoder collect at least κ blocks before attempting to decode. With this approach, the associated rate thresholds are $R, R\kappa/(\kappa+1), R\kappa/(\kappa+2), \ldots, R\kappa/M$. Hence, by choosing larger values of κ, one can increase the density of SNR thresholds.

IX. SIMULATIONS

Implicit in our analysis is the use of perfect base codes and ideal (maximum likelihood) decoding. In this section, we present simulations that further validate our rateless code design with practical coding and decoding.

In our simulations, we use as our base code the turbo code specified in the 3GPP LTE wireless standard [21], [22]. This parallel-concatenated convolutional code constructed from a pair of 8-state constituent encoders has a rate of 2/3 bits per complex symbol. This code is used in conjunction with the iterative turbo-decoding algorithm for which it was designed.

The base code is used in both 3- and 4-layer rateless constructions, corresponding to ceiling rates of $R = 2$ and $R = 8/3$ b/s/Hz, respectively. Moreover, there are a total of 6144 information bits per layer, corresponding to a block length of $N = 9216$ complex symbols.

Encoding proceeds as follows. Since the base code is not ideal, it has a bit-error rate that rolls off with the operating SNR. Let $\mathrm{SNR}_\circ(\epsilon)$ denote the SNR at which the base code achieves a bit-error rate of ϵ. Then, using a definition analogous to that used in Section VII-C, the efficiency of the base code is[9]

$$\eta_\circ(\epsilon) = \frac{R/L}{\log(1+\mathrm{SNR}_\circ(\epsilon))}.$$

Thus, in computing the gain matrix $\mathbf{G}$, we prescale the target rate, replacing R with $R/\eta_\circ(\epsilon)$. Note that as a result, $\mathbf{G}$ depends on the target rate and the base code properties only.

For the particular base code used in the simulations, the efficiencies are as given in Table III.

[9]One can equivalently measure the efficiency of the base code in terms of its gap to capacity at a particular target bit-error rate. However, our chosen measure is more natural when relating the efficiency of the base code to the rateless code constructed from it.

In our simulation, we decode not only from integer numbers of redundancy blocks, but also from noninteger numbers, corresponding to partial blocks. MMSE combining is applied on a symbol-by-symbol basis, in conjunction with our usual successive cancellation. In particular, when the number of incremental redundancy blocks m is noninteger, then the MMSE combiner for the first $N(m - \lfloor m \rfloor)$ symbols of the codeword in a given layer l is constructed from the submatrix $\mathbf{G}_{\lfloor m \rfloor+1,l}$, while the MMSE combiner for the remaining $N(1 + \lfloor m \rfloor - m)$ symbols of the codeword is constructed from the submatrices $\mathbf{G}_{\lfloor m \rfloor,l}$.

Following combining (and cancellation), turbo decoding is applied to the layer of interest, where the initial log-likelihood ratios are calculated treating the symbols as corrupted by Gaussian noise with variance determined by the effective SNR. This effective SNR is determined from the (reciprocal of the unbiased) mean-square error resulting from MMSE combining, taking into account the successive cancellation. Thus, when m is noninteger, the initial log-likelihood ratios take on one value for the symbols in the first part of the codeword, and a different value in the second part.

The overall efficiency η of the resulting rateless code, i.e., the fraction of capacity at which it operates, is a function of the number of incremental redundancy blocks m (or equivalently the realized SNR in the channel). We calculate η for the general case where m may be noninteger as follows. First, for a given value of m, the roll-off of the bit-error rate of the overall rateless code as a function of the SNR can be generated, where for each SNR value, the corresponding MMSE combiner with successive cancellation is used. As above, when m is noninteger two MMSE combiners are involved. The resulting bit error rate is averaged over both the N symbols within the codeword at every layer and the L layers, so that error propagation effects are taken into account. We then let $\mathrm{SNR}(m,\epsilon)$ denote the SNR at which the target bit-error rate ϵ is attained for this particular value of m, from which the efficiency of the rateless code is

$$\eta(m,\epsilon) = \frac{R/m}{\log(1+\mathrm{SNR}(m,\epsilon))} \tag{60}$$

where, again, we have used a notion of efficiency consistent with earlier definitions.

The resulting efficiency plots are depicted in Fig. 3. Several features are noteworthy. First, the efficiencies for $m = 1, 2, \ldots$ redundancy blocks are quite close to those of the base code shown in Table III; typically they are at most 2–3% lower. This suggests, at least for codes with few layers, that losses due to the rateless architecture itself, as well as the use of iterative decoding in the face of non-Gaussian noise from undecoded layers, are negligible in practice, and that good base codes will yield good rateless codes.

Second, the efficiencies do not vary significantly with the number of redundancy blocks m. Moreover, even when partial redundancy blocks are used, the efficiency does not deteriorate. This suggests that our rateless code constructions can operate over a much finer-grained set of rates than our design prescribed. However, it should be emphasized that this holds only when at least one full redundancy block is used. When less redundancy is used, Fig. 3 shows that efficiency falls off rapidly.

X. Extensions to Time-Varying Channels

The framework of this paper can be extended to time-varying channels in a variety of ways. As one example, the time-varying channel can be viewed as an instance of parallel channels, and thus a solution can be developed from a solution to the problem of rateless coding for parallel channels. Initial work in this direction is described in, e.g., [29]–[31], [33], though much remains to be understood about the performance limits of various constructions. Another approach is based on the observation that feedback about the past channel state can significantly simplify the problem of encoding for future transmissions [34]. It is this approach we describe here as an illustration of potential. In particular, we show that the natural generalization of our architecture is perfect (i.e., capacity-achieving), enabling the message to be recovered with the minimum possible number of blocks for the realized channel.

For the time-varying channel we consider, the observations take the form

$$\mathbf{y}_m = \beta_m \mathbf{x}_m + \mathbf{z}_m, \qquad m = 1, 2, \ldots \tag{61}$$

where the $\{\beta_m\}$ denote a sequence of complex channel gains. The β_m continue to be known *a priori* at the receiver but not at the transmitter.

The encoder transmits a message w by generating a sequence of incremental redundancy blocks $\mathbf{x}_1(w), \mathbf{x}_2(w), \ldots$. The receiver accumulates sufficiently many received blocks $\mathbf{y}_1, \mathbf{y}_2, \ldots$ to recover w. Immediately following the transmission of block $\mathbf{x}_m$, the encoder is notified of β_m. Thus, knowledge of $\beta_1, \ldots, \beta_m$ can be used in the construction of the redundancy block $\mathbf{x}_{m+1}(w)$.

In this context, a *perfect* rateless code is then one in which capacity is achieved for any number $m = 1, \ldots, M$ of redundancy blocks, i.e., whenever the (realized) channel gains are such that

$$R \le \sum_{m'=1}^{m} \log\left(1 + \frac{P}{\sigma^2}|\beta_{m'}|^2\right) \tag{62}$$

the message can be recovered with high probability.

In this development, for values of m such that the right side of (62) is less than R, it is convenient to define *target* channel gains α_{m+1} required for successful decoding once block $m+1$ is obtained. In particular, α_{m+1} is defined via

$$R = \log\left(1 + \frac{P}{\sigma^2}\alpha_{m+1}^2\right) + \sum_{m'=1}^{m} \log\left(1 + \frac{P}{\sigma^2}|\beta_{m'}|^2\right) \tag{63}$$

whenever $\alpha_m > |\beta_m|$.

Generalizing our construction for the time-invariant case, we first choose the range M, the ceiling rate R, the number of layers L, and finally the associated base codebooks $\mathcal{C}_1, \ldots, \mathcal{C}_L$. We assume *a priori* that the base codebooks all have equal rate R/L.

As with our time-invariant construction, the redundancy blocks $\mathbf{x}_1, \ldots, \mathbf{x}_M$ take the form (25). We emphasize that the mth row of $\mathbf{G}$, which constitutes the weights used in constructing the mth redundancy block from the L codewords, will in general be a function of the (realized) channel gains $\beta_1, \ldots, \beta_{m-1}$. Specifically, the mth row is designed for the channel gain sequence $\{\beta_1, \ldots, \beta_{m-1}, \alpha_m\}$, i.e., we substitute the target gain α_m for the (as yet unknown) channel gain β_m. Finally, in addition to the layered code structure, we continue to impose the constraint that the layered code be successively decodable.

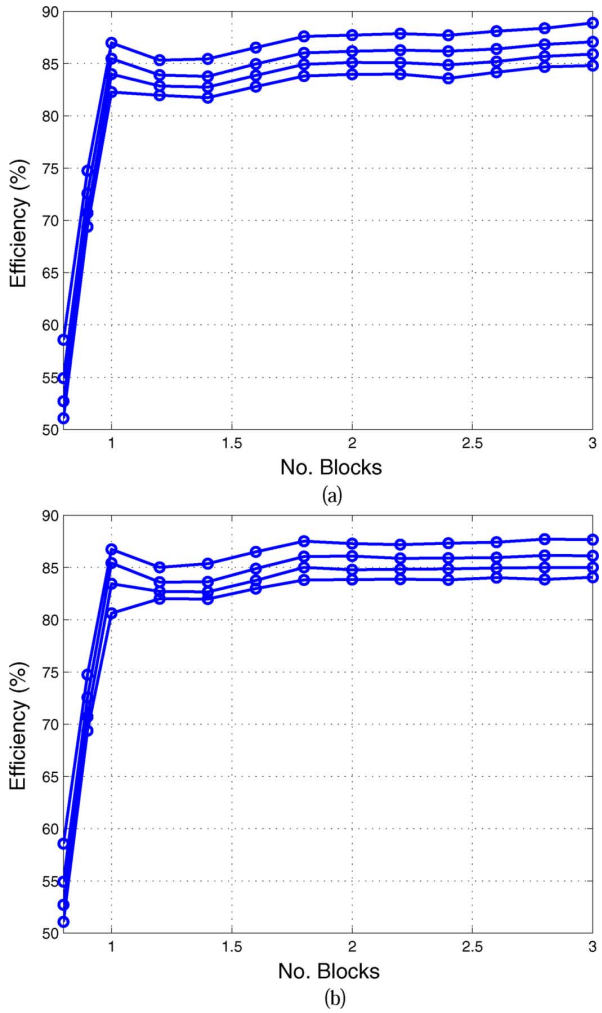

Fig. 3. Practical efficiencies achieved using a rateless construction in conjunction with rate 2/3 base code. The successively lower curves correspond to target bit-error rates of $10^{-2}, 10^{-3}, 10^{-4}$, and 10^{-5}, respectively. (a) 3-layers, 3-blocks (rate range: 2/3 to 2 b/s/Hz), (b) 4-layers, 4-blocks (rate range: 8/9 to 8/3 b/s/Hz).

Our aim is to select $\mathbf{G}$ so that the code is perfect as defined earlier. From the layered repetition encoding structure, we require as in the time-invariant development that the rows of $\mathbf{G}$ be orthogonal, while from the successive decoding constraint we have the requirement [cf. (36)] that

$$\frac{lR}{L} \le \log\det\left(\mathbf{I} + \frac{1}{\sigma^2}\mathbf{B}_m \mathbf{G}_{m,l} \mathbf{G}_{m,l}^{\dagger} \mathbf{B}_m^{\dagger}\right) \tag{64}$$

for all $l = 1, \ldots, L$ and $m = 1, \ldots, M$, with

$$\mathbf{B}_m = \mathrm{diag}(\beta_1, \ldots, \beta_{m-1}, \alpha_m). \tag{65}$$

With this model, in Appendix C we construct in closed form perfect rateless codes for the case of $M = 2$ redundancy blocks and $L = 3$ layers for rates in the range $R < \log(2+\sqrt{5}) \approx 2.08$ bits per complex symbol. This construction can be viewed as the time-varying natural generalization of that in Section IV. Establishing the existence of perfect rateless codes for larger values of M and/or L requires more effort. However, following an approach analogous to that used in corresponding development for

the time-invariant case in Section VII, we shown in Appendix D that in the limit of a large number of layers L, asymptotically perfect codes for all values of M are possible.

XI. Concluding Remarks

In this paper, motivated by hybrid ARQ requirements in wireless and related applications, our focus has been on the development of a lossless framework for transforming a code good for the AWGN channel at a single SNR into one good simultaneously at multiple SNRs. There are a variety of worthwhile directions for further research.

First, while beyond the scope of the present paper, a comparative evaluation of methods described herein relative to, for example, those described in Section II is likely to reveal additional insight, and uncover opportunities for further progress.

Second, while we have developed some preliminary results on the extension of our framework to time-varying channels, clearly this is just a beginning. For example, when $M > 2$, there is flexibility in the problem formulation, and thus in how the available degrees of freedom are allocated. As another example, one could consider other time-variation models, such as one that would allow β to vary deterministically so long as the pattern of variation is known in advance. Then, for one block the code would be designed for a gain of $[\alpha_{1,1}]$, for two blocks the target gains would be $[\alpha_{2,1}\ \alpha_{2,2}]$, for three blocks the gains would be $[\alpha_{3,1}\ \alpha_{3,2}\ \alpha_{3,3}]$, and so on. Still another example would involve the development of solutions for time-varying channels without requiring SNR feedback, either with or without a stochastic model for β.

Other worthwhile directions include more fully developing rateless constructions for the AWGN channel that allow decoding to begin at any received block, and/or to exploit an arbitrary subset of the subsequent blocks. Initial efforts in this direction include the faster-than-Nyquist constructions in [27], [29], [37] and the diagonal subblock layering approach described in [29].

Beyond the single-input, single-output (SISO) channel, many multiterminal and multiuser extensions are also of considerable interest. Examples of preliminary developments along these lines include the rateless space-time code constructions in [35], the rateless codes for multiple-access channels developed in [36], and the approaches to rateless coding for parallel channels examined in [29]–[31]. Indeed, such research may lead to efficient rateless orthogonal frequency-division multiplexing (OFDM) systems and efficient rateless multi-input, multi-output (MIMO) codes with wide-ranging applications.

Finally, extending the layered approach to rateless coding developed in this paper beyond the Gaussian channel is also a potentially rich direction for further research. A notable example would be the binary symmetric channel, where good rateless solutions remain elusive. Preliminary work in this direction is described in [32].

Appendix A
Perfect Rateless Solution for $L = M = 3$

Determining the set of solutions

$$\mathbf{G} = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} \tag{66}$$

to (36) when $L = M = 3$ as a function of the ceiling rate R is a matter of lengthy if routine algebra.

We begin by observing that any row or any column of $\mathbf{G}$ may be multiplied by a common phasor without changing $\mathbf{G}\mathbf{G}^{\dagger}$. Without loss of generality we may therefore take the first row and first column of $\mathbf{G}$ to be real and positive. Each $\mathbf{G}$ thus represents a set of solutions $\mathbf{D}_1\mathbf{G}\mathbf{D}_2$, where $\mathbf{D}_1$ and $\mathbf{D}_2$ are diagonal matrices in which the diagonal entries have modulus 1. The solutions in the set are equivalent for most engineering purposes and we shall therefore not distinguish them further.

We know that $\mathbf{G}$ must be a scaled unitary matrix, scaled so that the row and column norms are $\sqrt{P}$. Thus, if we somehow determine the first two rows of $\mathbf{G}$, there is always a choice for the third row: it's the unique vector orthogonal to the first two rows which meets the power constraint and which has first component real and positive. Conversely, it's easy to see that any appropriately scaled unitary matrix $\mathbf{G}$ that satisfies (36) for $m = 1$ and $m = 2$ (and all $l = 1, 2, 3$) necessarily satisfies (36) for $m = 3$. We may therefore without loss of generality restrict our attention to determining the set of solutions to the first two rows of $\mathbf{G}$; the third row comes "for free" from the constraint that $\mathbf{G}$ be a scaled unitary matrix.

Assume, again without loss of generality, that $\alpha_1^2 = 1$ and $\sigma^2 = 1$. Via (36), the first row of $\mathbf{G}$ (which controls the first redundancy block) must satisfy

$$R/3 = \log\left(1 + g_{11}^2\right) \tag{67}$$
$$2R/3 = \log\left(1 + g_{11}^2 + g_{12}^2\right) \tag{68}$$
$$3R/3 = \log\left(1 + g_{11}^2 + g_{12}^2 + g_{13}^2\right) \tag{69}$$

and must also satisfy the power constraint

$$P = g_{11}^2 + g_{12}^2 + g_{13}^2. \tag{70}$$

Thus

$$P = 2^R - 1 = x^6 - 1$$

and

$$g_{11}^2 = 2^{R/3} - 1 = x^2 - 1 \tag{71}$$
$$g_{12}^2 = 2^{R/3}(2^{R/3} - 1) = x^2(x^2 - 1) \tag{72}$$
$$g_{13}^2 = 2^{2R/3}(2^{R/3} - 1) = x^4(x^2 - 1) \tag{73}$$

where for convenience we have introduced the change of variables $x = 2^{R/6}$.

The first column of $\mathbf{G}$ (which controls the first layer of each redundancy block) is also straightforward. Via (29) with $m = 2$ and $m = 3$, we have

$$\alpha_2^2 = \frac{1}{x^3 + 1} \tag{74}$$
$$\alpha_3^2 = \frac{1}{x^4 + x^2 + 1}. \tag{75}$$

Using (36) for $l = 1$ and $m = 2$ yields

$$R/3 = \log\left(1 + \alpha_2^2\left(g_{11}^2 + g_{21}^2\right)\right). \tag{76}$$

Substituting the previously computed expressions (71) and (74) for g_{11}^2 and α_2^2 into (76) and solving for g_{21} yields

$$g_{21}^2 = x^3(x^2 - 1). \tag{77}$$

To solve for the second row of $\mathbf{G}$ we use (36) with $m = l = 2$ together with the requirement that the first and second rows be orthogonal. It is useful at this stage to switch to polar coordinates, i.e., $g_{22} = |g_{22}|e^{j\theta_1}$ and $g_{23} = |g_{23}|e^{j\theta_2}$.

Orthogonality of the first and second rows means that

$$0 = g_{11}g_{21} + g_{12}|g_{22}|e^{j\theta_1} + g_{13}|g_{23}|e^{j\theta_2}. \tag{78}$$

Complex conjugation is not needed here because the first row is real. The three terms in the above expression may be viewed as the legs of a triangle, so by the law of cosines

$$2g_{11}g_{21}g_{12}|g_{22}|\cos\theta_1 = g_{13}^2|g_{23}|^2 - g_{11}^2g_{21}^2 - g_{12}^2|g_{22}|^2. \tag{79}$$

We now use (36) with $m = l = 2$ to infer that

$$2^{2R/3} = x^4 = \det\left(\mathbf{I} + \alpha_2^2\mathbf{G}_{2,2}\mathbf{G}_{2,2}^\dagger\right). \tag{80}$$

To expand this expression, we compute

$$\mathbf{G}_{2,2}\mathbf{G}_{2,2}^\dagger = \begin{bmatrix} g_{11}^2 + g_{12}^2 & g_{11}g_{21} + g_{12}|g_{22}|e^{-j\theta_1} \\ (*) & g_{21}^2 + |g_{22}|^2 \end{bmatrix} \tag{81}$$

where $(*)$ is the complex conjugate of the upper right entry, from which we find

$$\begin{aligned}\det\left(\mathbf{I}+\alpha_2^2\mathbf{G}_{2,2}\mathbf{G}_{2,2}^\dagger\right) = &\alpha_2^4\left(g_{11}^2|g_{22}|^2 + g_{12}^2g_{21}^2\right. \\ &\left. -2g_{11}g_{21}g_{12}|g_{22}|\cos\theta_1\right) \\ &+ \alpha_2^2\left(g_{11}^2 + g_{12}^2 + g_{21}^2 + |g_{22}|^2\right) + 1.\end{aligned} \tag{82}$$

Substituting (79) into (82) to eliminate the cosine term and using (80) yields

$$\begin{aligned}x^4 = &\alpha_2^4\left(g_{11}^2|g_{22}|^2 + g_{12}^2g_{21}^2 - g_{13}^2|g_{23}|^2 + g_{11}^2g_{21}^2 + g_{12}^2|g_{22}|^2\right) \\ &+ \alpha_2^2\left(g_{11}^2 + g_{12}^2 + g_{21}^2 + |g_{22}|^2\right) + 1.\end{aligned} \tag{83}$$

Finally, substituting the expressions for $g_{11}^2, g_{12}^2, g_{13}^2, g_{21}^2$, and α_2^2 computed above, using the power constraint

$$|g_{23}|^2 = P - |g_{22}|^2 - g_{21}^2 \tag{84}$$

solving for $|g_{22}|^2$, and simplifying, we arrive at

$$|g_{22}|^2 = (x^5 + 1)(x - 1). \tag{85}$$

The power constraint (84) then immediately yields

$$|g_{23}|^2 = x(x^2 - 1). \tag{86}$$

The squared modulus of the entries of the last row of $\mathbf{G}$ follow immediately from the norm constraint on the columns:

$$g_{31}^2 = P - g_{21}^2 + g_{11}^2 = x^2(x^2 - x + 1)(x^2 - 1) \tag{87}$$

$$|g_{32}|^2 = P - g_{22}^2 - g_{12}^2 = x(x^3 + 1)(x - 1) \tag{88}$$

$$|g_{33}|^2 = P - g_{23}^2 - g_{13}^2 = (x^3 + 1)(x - 1). \tag{89}$$

This completes the calculation of the squared modulus of the entries of $\mathbf{G}$. In summary, we have shown that $\mathbf{G}$ has the form

$$\begin{aligned}\mathbf{G} = &\sqrt{x-1} \\ &\cdot\begin{bmatrix} \sqrt{x+1} & \sqrt{x^2(x+1)} & \sqrt{x^4(x+1)} \\ \sqrt{x^3(x+1)} & e^{j\theta_1}\sqrt{x^5+1} & e^{j\theta_2}\sqrt{x(x+1)} \\ \sqrt{x^2(x^3+1)} & e^{j\theta_3}\sqrt{x(x^3+1)} & e^{j\theta_4}\sqrt{x^3+1} \end{bmatrix}\end{aligned} \tag{90}$$

where $x = 2^{R/6}$.

We must now establish the existence of suitable $\theta_1, \ldots, \theta_4$. To resolve this question it suffices to consider the consequences of the orthogonality constraint (78) on θ_1 and θ_2. As remarked at the start of this section, the last row of $\mathbf{G}$ and hence θ_3 and θ_4 come "for free" once we have the first two rows of $\mathbf{G}$.

Substituting the expressions for $|g_{ml}|^2$ determined above into (78) and canceling common terms yields

$$0 = \sqrt{x} + e^{j\theta_1}\sqrt{x^4 - x^3 + x^2 - x + 1} + e^{j\theta_2}\sqrt{x^3}. \tag{91}$$

The right-hand side is a sum of three phasors of predetermined magnitude, two of which can be freely adjusted in phase. In geometric terms, the equation has a solution if we can arrange the three complex phasors into a triangle, which is possible if and only if the longest side of the triangle is no longer than the sum of the lengths of the shorter sides. The resulting triangle is unique (up to complex conjugation of all the phasors). Now, the middle term of (91) grows faster in x than the others, so for large x we cannot possibly construct the desired triangle. A necessary condition for a solution is thus

$$\sqrt{x} + \sqrt{x^3} \geq \sqrt{x^4 - x^3 + x^2 - x + 1} \tag{92}$$

where equality can be shown (after some manipulation) to hold at the largest root of $x^2 - x + 1$, i.e., at $x = (3 + \sqrt{5})/2$, or equivalently $R = 6\log_2 x = 6\log_2(3 + \sqrt{5}) - 6$. It becomes evident by numerically plotting the quantities involved that this necessary condition is also sufficient, i.e., a unique solution to (91) exists for all values of x in the range $1 < x \leq (3 + \sqrt{5})/2$ and no others. Establishing this fact algebraically is an unrewarding though straightforward exercise.

A relatively compact formula for θ_1 may be found by applying the law of cosines to (91), whence

$$\cos(\pi - \theta_1) = \frac{x^4 - 2x^3 + x^2 + 1}{2\sqrt{x(x^4 - x^3 + x^2 - x + 1)}}. \tag{93}$$

Similar formulas may be derived for θ_2, θ_3, and θ_4.

Appendix B
Power Allocation

The power allocation satisfying the property (50) can be obtained as the solution to a different but closely related rateless code optimization problem. Specifically, let us retain the block structuring and layering of the code of Section VII-A, but instead of using repetition and dithering in the construction, let us consider a code where the codebooks in a given layer are *independent* from block to block. While such a code is still successively decodable, it does not retain other characteristics

that make decoding possible with low complexity. However, the complexity characteristic is not of interest. What does matter to us is that the per-layer, per-block SNRs that result from a particular power allocation will be identical to those of the code of Section VII-A for the same power allocation. Thus, in tailoring our code in this Appendix to meet (50), we simultaneously ensure our code of Section VII-A will as well.

We begin by recalling a useful property of layered codes in general that we will apply. Consider an AWGN channel with gain β and noise z of variance σ^2, and consider an L-layer block code that is successively decodable. If the constituent codes are capacity-achieving i.i.d. Gaussian codes, and MMSE successive cancellation is used, then the overall code will be capacity achieving. More specifically, for any choice of powers p_l for layers $l = 1, 2, \ldots, L$ that sum to the power constraint P, the associated rates I_l for these layers will sum to the corresponding capacity $\log(1+|\beta|^2 P/\sigma^2)$. Equivalently, for any choice of rates I_l that sum to capacity, the associated powers p_l will sum to the corresponding power constraint. In this latter case, any rate allocation that yield powers that are all nonnegative is a valid one.

To see this, let the relevant codebooks for the layers be $\tilde{\mathcal{C}}_1, \ldots, \tilde{\mathcal{C}}_L$, and let the overall codeword be denoted

$$\tilde{c} = \tilde{c}_1 + \ldots + \tilde{c}_L \tag{94}$$

where the $\tilde{c}_l \in \tilde{\mathcal{C}}_l$ are independently selected codewords drawn for each layer. The overall code rate is the sum of the rates of the individual codes. The overall power of the code is $P = p_1 + \ldots + p_L$.

From the mutual information decomposition

$$I(\tilde{c}; y) = \sum_{l=1}^{L} I_l \tag{95}$$

where

$$I_l = I\left(\tilde{c}_l; \tilde{c}_1 + \ldots + \tilde{c}_L + z \,\middle|\, \tilde{c}_{l+1}^L\right)$$

with $\tilde{c}_{l+1}^L = (\tilde{c}_{l+1}, \tilde{c}_{l+2}, \ldots, \tilde{c}_L)$, we see that the overall codebook power constraint P can be met by apportioning power to layers in any way desired, so long as $p_1 + \ldots + p_L = P$. Since the undecoded layers are treated as noise, the maximum codebook rate for the lth layer is then

$$I_l = \log(1 + \mathrm{SNR}_l) \tag{96}$$

where

$$\mathrm{SNR}_l = \frac{|\beta|^2 p_l}{|\beta|^2 p_1 + |\beta|^2 p_2 + \cdots + |\beta|^2 p_{l-1} + \sigma^2} \tag{97}$$

is the effective SNR when decoding the lth layer. Straightforward algebra, which amounts to a special-case recalculation of (95), confirms that $I_1 + \ldots + I_L = \log(1 + |\beta|^2 P/\sigma^2)$ for any selection of powers $\{p_l\}$.

Alternatively, instead of selecting per-layer powers and computing corresponding rates, one can select per-layer rates and compute the corresponding powers. The rates $\{I_l\}$ for each level may be set in any way desired so long as the total rate $I_1 + \ldots + I_L$ does not exceed the channel capacity $\log(1+|\beta|^2 P/\sigma^2)$. The required powers $\{p_l\}$ may then be found using (96) and (97) recursively for $l = 1, \ldots, L$. There is no need to verify the power constraint: it follows from (95) that the powers computed in this way sum to P. Thus it remains only to check that the $\{p_l\}$ are all nonnegative to ensure that the rate allocation is a valid one.

We now apply this insight to our rateless context. The target ceiling rate for our rateless code is R, and, as before, α_m, $m = 1, 2, \ldots$, denotes the threshold channel gains as obtained via (29).

Comparing (50) with (96) and (97) reveals that (50) can be rewritten as

$$R_l = \sum_{m'=1}^{m} I_{m',l}(\alpha_m) \tag{98}$$

for all $l = 1, 2, \ldots, L$ and $m = 1, 2, \ldots$, where

$$R_l = R/L \tag{99}$$

and $I_{m',l}(\alpha_m)$ is the mutual information in layer l from block m' when the realized channel gain is α_m. Thus, meeting (50) is equivalent to finding powers $p_{m',l}$ for each code block m' and layer l so that for the given rate allocation R_l (a) the powers are nonnegative, (b) the power constraint is met, and (c) when the channel gain is α_m, the mutual information accumulated at the lth layer after receiving code blocks $1, 2, \ldots, m$ equals R_l.

Since the power constraint is automatically satisfied by any assignment of powers that achieves the target rates, it suffices to establish that (98) has a solution with nonnegative per-layer powers.

The solution exists and is unique, as can be established by induction on m. Specifically, for $m = 1$ the rateless code is an ordinary layered code and the powers $p_{1,1}, \ldots, p_{1,L}$ may be computed recursively from [cf. (98)]

$$R_l = \sum_{m'=1}^{m} \log(1 + \mathrm{SNR}_{m',l}(\alpha_m)) \tag{100}$$

with $\mathrm{SNR}_{m,l}(\alpha_m)$ as given in (49) for $l = 1, \ldots, L$.

For the induction hypothesis, assume we have a power assignment for the first m blocks that satisfies (100). To find the power assignment for the $(m+1)$st block, observe that when the channel gain decreases from α_m to α_{m+1} the per-layer mutual information of every block decreases. A nonnegative power must be assigned to every layer in the $(m+1)$st code block to compensate for the shortfall.

The mutual information shortfall in the lth layer is

$$\Delta_{m+1,l} = R_l - \sum_{m'=1}^{m} \log(1 + \mathrm{SNR}_{m',l}(\alpha_{m+1})) \tag{101}$$

and the power $p_{m+1,l}$ needed to make up for this shortfall is the solution to

$$\Delta_{m+1,l} = \log(1 + \mathrm{SNR}_{m+1,l}(\alpha_{m+1})) \tag{102}$$

viz.,

$$p_{m+1,l} = (2^{2\Delta_{m+1,l}} - 1) \cdot \left(p_{m+1,1} + \cdots + p_{m+1,l-1} + \frac{\sigma_{m+1}^2}{\alpha_{m+1}^2}\right). \tag{103}$$

This completes the induction. Perhaps counter to intuition, even if the per-layer rates $R_1, \ldots, R_L$ are set equal, the per-layer

TABLE IV
PER-LAYER POWER ASSIGNMENTS $p_{m,l}$ AND CHANNEL GAIN THRESHOLDS α_m FOR THE INITIAL BLOCKS OF AN $L = 4$ LAYER RATELESS CODE WITH TOTAL POWER $P = 255$, NOISE VARIANCE $\sigma^2 = 1$, AND PER-LAYER RATE $R/L = 1$ B/S/HZ

	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$
gain (dB)	0.00	-12.30	-16.78	-19.29	-20.99
$l=1$	3.00	40.80	48.98	55.77	58.79
$l=2$	12.00	86.70	61.21	60.58	61.65
$l=3$	48.00	86.70	81.32	71.48	67.50
$l=4$	192.00	40.80	63.48	67.16	67.06

shortfalls $\Delta_{m+1,1}, \ldots, \Delta_{m+1,L}$ will not be equal. Thus, within a layer the effective SNR and mutual information will vary from block to block.

Equations (101) and (103) are easily evaluated numerically. An example is given in Table IV.[10]

Finally, since this result holds regardless of the choice of the constituent R_l, it will hold for the particular choice (99), whence (50).

APPENDIX C
PERFECT $L = 3, M = 2$ RATELESS SOLUTION FOR TIME-VARYING CHANNEL

As the simplest example, for the case of $M = 2$ redundancy blocks and $L = 3$ layers the constraints (64) can be met, i.e., a perfect rateless code is possible provided R is not too large.

In this case, we determine our gain matrix

$$\mathbf{G} = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \end{bmatrix} \tag{104}$$

as a function of the ceiling rate R, where the second row also depends on the realized channel gain β_1 experienced by the first incremental redundancy block.

As in the time-invariant case, we may without loss of generality take the first row and column to be real and nonnegative. Assume, also without loss of generality, that $\alpha_1 = 1$ and $\sigma^2 = 1$. Then the first row of $\mathbf{G}$, which corresponds to the first redundancy block, is computed exactly as in the time-invariant case. In particular, from (64) with $m = 1$, it must satisfy

$$R/3 = \log\left(1 + g_{11}^2\right) \tag{105}$$
$$2R/3 = \log\left(1 + g_{11}^2 + g_{12}^2\right) \tag{106}$$
$$3R/3 = \log\left(1 + g_{11}^2 + g_{12}^2 + g_{13}^2\right) \tag{107}$$

together with the power constraint

$$P = g_{11}^2 + g_{12}^2 + g_{13}^2. \tag{108}$$

Thus, with $x \triangleq 2^{R/6}$, we have

$$P = 2^R - 1 = x^6 - 1 \tag{109}$$

[10]If one were aiming to use a rateless code of the type described in Section VII in practice, in calculating a power allocation one should take into account the gap to capacity of the particular base code being used. Details of this procedure for the case of perfect rateless codes are given as part of the description of the simulations in Section IX. For the case of near perfect codes, the corresponding procedure is described in [28].

$$g_{11}^2 = 2^{R/3} - 1 = x^2 - 1 \tag{110}$$
$$g_{12}^2 = 2^{R/3}(2^{R/3} - 1) = x^2(x^2 - 1) \tag{111}$$
$$g_{13}^2 = 2^{2R/3}(2^{R/3} - 1) = x^4(x^2 - 1). \tag{112}$$

The derivation now departs from the time-invariant case. Recall that β_1 is the realized channel gain for the first block. A second redundancy block is thus needed when $|\beta_1| < \alpha_1$. The target gain α_2 is the solution to [cf. (63)]

$$R = \log(1 + P|\beta_1|^2) + \log\left(1 + P\alpha_2^2\right) \tag{113}$$

which is

$$\alpha_2^2 = \frac{1 - |\beta_1|^2}{1 + P|\beta_1|^2}. \tag{114}$$

Using (64) for $m = 2$ and $l = 1$ yields

$$R/3 = \log\left(1 + |\beta_1|^2 g_{11}^2 + \alpha_2^2 g_{21}^2\right). \tag{115}$$

Substituting the previously computed expressions (110) and (114) for g_{11}^2 and α_2^2 and solving for g_{21} yields

$$g_{21}^2 = (x^2 - 1)(1 + P|\beta_1|^2). \tag{116}$$

As in the time-invariant case, to solve for the rest of the second row of $\mathbf{G}$ we use (64) with $m = l = 2$ together with the requirement that the first and second rows be orthogonal. It is useful at this stage to switch to polar coordinates, i.e., $g_{22} = |g_{22}|e^{j\theta_1}$ and $g_{23} = |g_{23}|e^{j\theta_2}$.

Orthogonality of the first and second rows means that

$$0 = g_{11}g_{21} + g_{12}|g_{22}|e^{j\theta_1} + g_{13}|g_{23}|e^{j\theta_2}. \tag{117}$$

The three terms in the above expression may be viewed as the legs of a triangle, so by the law of cosines

$$2g_{11}g_{21}g_{12}|g_{22}|\cos\theta_1 = g_{13}^2|g_{23}|^2 - g_{11}^2 g_{21}^2 - g_{12}^2|g_{22}|^2. \tag{118}$$

We now use (64) with $m = l = 2$ to infer that

$$2^{2R/3} = x^4 = \det(\mathbf{I} + \mathrm{diag}\{|\beta_1|^2|, \alpha_2^2\}\mathbf{G}_{2,2}\mathbf{G}_{2,2}^\dagger). \tag{119}$$

To expand this expression, we compute

$$\mathbf{G}_{2,2}\mathbf{G}_{2,2}^\dagger = \begin{bmatrix} g_{11}^2 + g_{12}^2 & g_{11}g_{21} + g_{12}|g_{22}|e^{-j\theta_1} \\ (*) & g_{21}^2 + |g_{22}|^2 \end{bmatrix} \tag{120}$$

where $(*)$ is the complex conjugate of the upper right entry, from which we find

$$\begin{aligned} \det\left(\mathbf{I} + \mathrm{diag}\left\{|\beta_1|^2|, \alpha_2^2\right\}\mathbf{G}_{2,2}\mathbf{G}_{2,2}^\dagger\right) &= |\beta_1|^2\alpha_2^2\left(g_{11}^2|g_{22}|^2 + g_{12}^2 g_{21}^2 - 2g_{11}g_{21}g_{12}|g_{22}|\cos\theta_1\right) \\ &\quad + |\beta_1|^2\left(g_{11}^2 + g_{12}^2\right) + \alpha_2^2\left(g_{21}^2 + |g_{22}|^2\right) + 1. \end{aligned} \tag{121}$$

Substituting (118) into (121) and using (119) yields

$$\begin{aligned} x^4 &= |\beta_1|^2\alpha_2^2\left(g_{11}^2|g_{22}|^2 + g_{12}^2 g_{21}^2 - g_{13}^2|g_{23}|^2\right. \\ &\qquad \left. + g_{11}^2 g_{21}^2 + g_{12}^2|g_{22}|^2\right) \qquad (122) \\ &\quad + |\beta_1|^2\left(g_{11}^2 + g_{12}^2\right) + \alpha_2^2\left(g_{21}^2 + |g_{22}|^2\right) + 1. \qquad (123) \end{aligned}$$

Finally, substituting the expressions for $g_{11}^2, g_{12}^2, g_{13}^2, g_{21}^2$, and α_2^2 computed above, using the power constraint

$$|g_{23}|^2 = P - |g_{22}|^2 - g_{21}^2 \tag{124}$$

solving for $|g_{22}|^2$, and simplifying terms, we arrive at

$$\begin{aligned}|g_{22}|^2 = \frac{x^2-1}{1+(x^6-1)|\beta_1|^2} \\ \cdot (x^2 + |\beta_1|^2(x^{10}+x^8-x^6-x^4-x^2+1) \\ - |\beta_1|^4(x^6-1)^2).\end{aligned} \tag{125}$$

Evidently, a necessary condition for the existence of a solution for $\mathbf{G}$ is that $g_{21}^2 + |g_{22}|^2 < P$. It can be shown that the sum of the powers on the first two layers is maximized when $|\beta_1| \to 1$, and then the necessary condition simplifies to

$$2^{R+1} - 2^{2R/3+1} < 2^R - 1 \tag{126}$$

which may be shown to hold for all $R < \log(2+\sqrt{5}) \approx 2.08$ bits per complex symbol.

The final step—a straightforward exercise, the details of which we omit—is to apply the triangle inequality to (117) to prove that the required triangle exists, and thus the phases θ_1 and θ_2.

Appendix D
Near-Perfect Rateless Codes for Time-Varying Channels

Our construction is a slight generalization of the corresponding construction in Section VII for time-invariant channels. First, we fix M, R, L, and the associated codebooks $\mathcal{C}_l, \ldots, \mathcal{C}_L$ each of rate R'/L for some $R' < R$ to be determined. Using $\mathsf{c}_l(n)$ and $\mathsf{x}_m(n)$ to denote the nth elements of codeword $\mathbf{c}_l$ and redundancy block $\mathbf{x}_m$, respectively, we again have (38).

Power Allocation

As in the corresponding development for the time-invariant case in Section B, a suitable power allocation for our construction is obtained as that which is optimum for a slightly different construction, which we now develop. In this section, different (independent) codebooks are used for different redundancy blocks, and we take $\mathbf{G}(n)$ to be independent of n, so that $\mathbf{G}(n) = \mathbf{P}$, where $\mathbf{P}$ is as given in (40).

The mutual information in the lth layer of the mth block is then

$$I_{m,l} = \log(1 + \mathrm{SNR}_{m,l}(\beta_m)) \tag{127}$$

where

$$\mathrm{SNR}_{m,l}(\beta_m) = \frac{|\beta_m|^2 p_{m,l}}{|\beta_m|^2(p_{m,1} + \ldots + p_{m,l-1}) + 1} \tag{128}$$

is the associated per-layer SNR experienced during successive decoding.

We now obtain the elements of $\mathbf{P}$ recursively. We proceed from the first block $m = 1$ to block M, where in each block m we start by determining $P_{m,1}$ and proceed up through $P_{m,L}$. By definition of α_1, we have

$$\log\left(1 + \alpha_1^2 \sum_{l=1}^{L} P_{1,l}\right) = R.$$

Viewing the layering as superposition coding for a multi-access channel, it is clear that any rate vector is achievable as long as its sum-rate is R. We may therefore obtain an equal rate per layer by taking $P_{1,1}, \ldots, P_{1,L}$ such that

$$\log\left(1 + P_{1,l}\alpha_1^2\right) = R/L, \qquad l = 1, \ldots, L. \tag{129}$$

Upon receiving knowledge of $|\beta_1|$ we proceed to determine the power allocation for block $m = 2$. More generally, suppose the power allocations through block $m-1$ have been determined and we have now acquired channel state knowledge through β_{m-1}. To determine the allocation for block m, we first compute the mutual information shortfall in layer l as

$$\Delta_{m,l} = \frac{R}{L} - \sum_{m'=1}^{m-1} \log(1 + \mathrm{SNR}_{m',l}(\beta_{m'})). \tag{130}$$

By the induction hypothesis, had the realized channel gain been $|\beta_{m-1}| = \alpha_{m-1}$, then $\Delta_{m,l}$ would be zero for all $l = 1, \ldots, L$. Now since we have $|\beta_{m-1}| < \alpha_{m-1}$, clearly the shortfall is positive for all layers. Also, by definition of α_m, we also have

$$\Delta_m = \sum_{l=1}^{L} \Delta_{m,l} = \log\left(1 + P\alpha_m^2\right). \tag{131}$$

We then solve for $p_{m,1}, \ldots, p_{m,L}$, in order, via

$$\log(1 + \mathrm{SNR}_{m,l}(\alpha_m)) = \Delta_{m,l}. \tag{132}$$

The resulting power allocation ensures that the aggregate mutual information per layer is at least R/L if $|\beta_m| > \alpha_m$ when i.i.d. Gaussian codebooks for all layers and blocks. However, we wish to use the same set of L codebooks for all redundancy blocks, to keep decoding complexity low. We return to this problem next, but in doing so will exploit this power allocation.

Encoding

In our construction we restrict our attention to an encoding of the form described in Section VII-A. In particular, the $\mathbf{G}(n)$ are of the form (39) with (40) and (41), with the $\mathsf{d}_{m,l}(n)$ all i.i.d. random variables in m, l, and n, and drawn independently of all other random variables, including noises, messages, and codebooks. As before, it is sufficient for $\mathsf{d}_{m,l}(n)$ to take on only values ± 1, and with equal probability.

Decoding

Decoding proceeds in a manner analogous to that described in Section VII-B for the time-invariant case. In particular, since $\mathbf{G}(n)$ is drawn i.i.d., the overall channel is i.i.d., and thus we may express the channel model in terms of an arbitrary individual element in the block. Specifically, assume that the channel gain for

block m is the minimal required $\beta_m = \alpha_m$, then our received symbol can be expressed as [cf. (42)]

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \mathbf{B}_m \mathbf{G} \begin{bmatrix} c_1 \\ \vdots \\ c_L \end{bmatrix} + \begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix}$$

where $\mathbf{G} = \mathbf{P} \odot \mathbf{D}$, with $\mathbf{G}$ denoting the arbitrary element in the sequence $\mathbf{G}(n)$, and where $y_{m'}$ is the corresponding received symbol from redundancy block m' (and similarly for $c_{m'}, z_{m'}, \mathbf{D}$).

As in the time-invariant case, it is sufficient to employ successive cancellation decoding with simple maximal ratio combining (MRC) of the redundancy blocks. In this case, the effective SNR at which this lth layer is decoded from m blocks via such MRC decoding is given by [cf. (48)]

$$\mathrm{SNR}_{\mathrm{MRC}} = \sum_{m'=1}^{m} \mathrm{SNR}_{m',l}(\beta_m) \tag{133}$$

with $\mathrm{SNR}_{m',l}(\beta_m)$ is as given in (128).

Efficiency Analysis

To show that the resulting scheme is asymptotically perfect, we first note that when random dither encoding, MRC decoding, and capacity-achieving base codes are used, the mutual information $I'_{m,l}$ satisfies [cf. (55)]

$$I'_{m,l} \geq \log(1 + \mathrm{SNR}_{\mathrm{MRC}}(\beta_m)) \tag{134}$$

with $\mathrm{SNR}_{\mathrm{MRC}}(\beta_m)$ as in (133).

Again the efficiency of the scheme depends on the choice of power allocation matrix (40). Recall that we may further bound $I'_{m,l}$ for all m by (56). Thus, if we choose the rate R''/L of the base code in each layer to be (57) then (56) ensures decodability after m blocks are received when the channel gain satisfies $|\beta_m| \geq \alpha_m$, as required. Moreover, the efficiency R''/R can be made as close as desired to one by taking L sufficiently large.

ACKNOWLEDGMENT

The authors thank the reviewers and associate editor for their helpful comments, which led to a number of improvements in the manuscript. We also acknowledge the efforts of Yuval Har-Zion, whose project at Tel Aviv University provided the simulation results depicted in Fig. 3.

REFERENCES

[1] G. Caire and D. Tuninetti, "The throughput of hybrid-ARQ protocols for the Gaussian collision channel," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1971–1988, Jul. 2001.

[2] D. Chase, "Code combining—A maximum-likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Trans. Commun.*, vol. 33, no. 5, pp. 385–393, May 1985.

[3] J. Ha, J. Kim, and S. W. McLaughlin, "Rate-compatible puncturing of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2824–2836, Nov. 2004.

[4] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389–400, Apr. 1988.

[5] T. Ji and W. Stark, "Rate-adaptive transmission over correlated fading channels," *IEEE Trans. Commun.*, vol. 53, no. 10, pp. 1663–1670, Oct. 2005.

[6] S. Lin, D. Costello, and M. Miller, "Automatic-repeat-request error-control schemes," *IEEE Commun. Mag.*, vol. 22, no. 12, pp. 5–17, Dec. 1984.

[7] D. M. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Trans. Inf. Theory*, vol. 20, no. 3, pp. 388–389, May 1974.

[8] D. N. Rowitch and L. B. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo (RCPT) codes," *IEEE Trans. Commun.*, vol. 48, no. 6, pp. 948–959, Jun. 2000.

[9] S. Sesia, G. Caire, and G. Vivier, "Incremental redundancy hybrid ARQ schemes based on low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1311–1321, Aug. 2004.

[10] N. Shulman and M. Feder, "Static broadcasting," in *Proc. Int. Symp. Inf. Theory (ISIT)*, Sorrento, Italy, Jun. 2000, p. 23.

[11] N. Shulman, "Universal channel coding," Ph.D. thesis, Tel-Aviv Univ., Ramat Aviv, Israel, 2004.

[12] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, Jun. 2006.

[13] J. W. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," *IEEE J. Select. Areas Commun.*, vol. 20, no. 5, pp. 1528–1540, Oct. 2002.

[14] M. Luby, "Information additive code generator and decoder for communication systems," U.S. U.S. Pat. No. 6307487, Oct. 23, 2001.

[15] E. Soljanin, N. Varnica, and P. Whiting, "'Punctured vs rateless codes for hybrid ARQ," in *Proc. Int. Inf. Theory Workshop (ITW)*, Punta del Este, Uruguay, Mar 2006, pp. 155–159.

[16] O. Etesami and A. Shokrollahi, "Raptor codes on binary memoryless symmetric channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 5, pp. 2033–2051, May 2006.

[17] R. Palanki and J. S. Yedidia, "Rateless codes on noisy channels," in *Proc. Int. Symp. Inf. Theory (ISIT)*, Chicago, IL, Jun. 2004, p. 37.

[18] A. S. Barbulescu and S. S. Pietrobon, "Rate compatible turbo codes," *Electron. Lett.*, pp. 535–536, 1995.

[19] R. Mantha and F. R. Kschischang, "A capacity approaching hybrid ARQ scheme using Turbo codes," in *Proc. GLOBECOM*, 1999, pp. 2341–2345.

[20] J. Li and K. R. Narayanan, "Rate-compatible low density parity check codes for capacity-approaching ARQ scheme in packet data," in *Int. Conf. Commun., Internet, Inf. Technol. (CIIT)*, St. Thomas, U.S. Virgin Islands, Nov. 2002.

[21] 3rd Generation Partnership Project (3GPP), "Evolved universal terrestrial radio access (E-UTRA); physical channels and modulation," 3rd Generation Partnership Project 2009 [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36211.htm, 3GPP TS 36.211, [Online]. Available:

[22] 3rd Generation Partnership Project (3GPP), "Evolved universal terrestrial radio access (E-UTRA); multiplexing and channel coding," 3rd Generation Partnership Project 2009 [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36212.htm, 3GPP TS 36.212,. [Online]. Available:

[23] M. Doettling, T. Grundler, and A. Seeger, "Incremental redundancy and bit-mapping techniques for high-speed downlink packet access," in *Proc. Global Commun. Conf. (GLOBECOM)*, San Francisco, CA, Dec. 2003, pp. 908–912.

[24] C. Wengerter, A. G. E. von Elbwart, E. Seidel, G. Velev, and M. P. Schmitt, "Advanced hybrid ARQ technique employing a signal constellation rearrangement," in *Proc. Vehic. Technol. Conf. (VTC)*, Vancouver, Canada, Sep. 2002, vol. 4, pp. 2002–2006.

[25] R. J. Barron, C. K. Lo, and J. M. Shapiro, "Global design methods for Raptor codes using binary and higher-order modulations," in *Proc. Military Commun. Conf. (MILCOM)*, Boston, MA, Oct. 2009.

[26] U. Erez, G. W. Wornell, and M. D. Trott, "Rateless coding and perfect rate-compatible codes for Gaussian channels," in *Proc. Int. Symp. Inf. Theory (ISIT)*, Seattle, WA, Jul. 2006, pp. 528–532.

[27] U. Erez, G. W. Wornell, and M. D. Trott, "Coding for faster-than-Nyquist signaling: The merits of a regime change," in *Proc. Allerton Conf. Commun., Contr., Computing*, Monticello, IL, Sep. 2004.

[28] J. M. Shapiro, R. J. Barron, and G. W. Wornell, "Practical layered rateless codes for the Gaussian channel: Power allocation and implementation," in *Int. Workshop Signal Processing Adv. Wireless Commun. (SPAWC)*, Helsinki, Finland, Jun. 2007.

[29] M. Shanechi, "Universal codes for parallel Gaussian channels," S.M. Thesis, Massachusetts Inst. Technol., Cambridge, MA, 2006.

[30] M. Shanechi, U. Erez, and G. W. Wornell, "On universal coding for parallel Gaussian channels," in *Proc. Int. Zurich Seminar Commun.*, Zurich, Switzerland, Mar. 2008.

[31] M. Shanechi, U. Erez, G. W. Wornell, and K. P. Boyle, "Time-invariant rateless codes for MIMO channels," in *Proc. Int. Symp. Inf. Theory*, Toronto, ON, Canada, Jul. 2008.
[32] J. Jiang and K. R. Narayanan, "Multilevel coding for channels with non-uniform inputs and rateless transmission over the BSC," in *Proc. Int. Symp. Inf. Theory (ISIT)*, Seattle, WA, Jul. 2006, pp. 518–521.
[33] M. Shanechi, U. Erez, and G. W. Wornell, "Rateless coding for MIMO channels," in *Proc. Int. Global Commun. Conf. (GLOBECOM)*, New Orleans, LA, Nov. 2008.
[34] U. Erez, M. D. Trott, and G. W. Wornell, "An efficient ARQ scheme with SNR feedback," in *Proc. Int. Zurich Seminar Commun.*, Zurich, Switzerland, Mar. 2008.
[35] U. Erez, G. W. Wornell, and M. D. Trott, "Rateless space-time coding," in *Proc. Int. Symp. Inf. Theory (ISIT)*, Adelaide, Australia, Sep. 2005, pp. 1937–1941.
[36] U. Niesen, U. Erez, D. Shah, and G. W. Wornell, "Rateless codes for the Gaussian multiple-access channel," in *Proc. IEEE GLOBECOM*, Nov. 2006.
[37] U. Erez and G. W. Wornell, "A super-Nyquist architecture for reliable underwater acoustic communication," in *Proc. Allerton Conf. Commun., Contr., Computing*, Monticello, IL, Sep. 2011.

Uri Erez (M'03) was born in Tel-Aviv, Israel, on October 27, 1971. He received the B.Sc. degree in mathematics and physics and the M.Sc. and Ph.D. degrees in electrical engineering from Tel-Aviv University, Israel, in 1996, 1999, and 2003, respectively.

During 2003–2004, he was a Postdoctoral Associate at the Signals, Information and Algorithms Laboratory at the Massachusetts Institute of Technology (MIT), Cambridge, MA. Since 2005, he has been with the Department of Electrical Engineering-Systems at Tel-Aviv University. His research interests are in the general areas of information theory and digital communication.

Dr. Erez is currently an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY.

Mitchell D. Trott (S'90–M'92–SM'03–F'11) received the B.S. and M.S. degrees in systems engineering from Case Western Reserve University, Cleveland, OH, in 1987 and 1988, respectively, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1992.

He was an Associate Professor in the Department of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology, Cambridge, MA, from 1992 until 1998, and Director of Research at ArrayComm, Inc., San Jose, CA, from 1997 through 2002. He is now a Distinguished Technologist at Hewlett-Packard Laboratories, Palo Alto, CA, where he leads the Seamless Collaboration project. His research interests include streaming media systems, multimedia collaboration, multi-user and wireless communication, and information theory.

Gregory W. Wornell (S'83–M'91–SM'00–F'04) received the B.A.Sc. degree from the University of British Columbia, Vancouver, BC, Canada, and the S.M. and Ph.D. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, MA, all in electrical engineering and computer science, in 1985, 1987, and 1991, respectively.

Since 1991, he has been on the faculty at MIT, where he is Professor of Electrical Engineering and Computer Science (EECS), leads the Signals, Information, and Algorithms Laboratory in the Research Laboratory of Electronics, chairs the graduate area of Communication, Control, and Signal Processing in EECS, and is Co-Director of the Center for Wireless Networking. He has held visiting appointments at the former AT&T Bell Laboratories, Murray Hill, NJ, the University of California, Berkeley, CA, and Hewlett-Packard Laboratories, Palo Alto, CA. His research interests and publications span the areas of signal processing, digital communication, and information theory, and include algorithms and architectures for wireless and sensor networks, broadband systems, and multimedia environments.

Dr. Wornell has been involved in both the Information Theory and Signal Processing Societies of the IEEE in a variety of capacities, and maintains a number of close industrial relationships and activities. He has won a number of awards for his research and teaching.