Rateless Coding and Perfect Rate-Compatible Codes for Gaussian Channels

Uri Erez
Dept. EE - Systems, Tel Aviv University
Tel Aviv, Israel
Email: uri@eng.tau.ac.il

Mitchell D. Trott
Hewlett-Packard Laboratories
Palo Alto, CA
Email: mitchell.trott@hp.com

Gregory W. Wornell
Dept. EECS, MIT
Cambridge, MA
Email: gww@mit.edu

Abstract— A rateless code, or a rate-compatible family of codes, has the property that the higher rate codes have codewords that 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. As an illustration of our framework, we design a practical three-rate code family. We further demonstrate that a rich set of perfect or near-perfect rateless codes may be found via numerical optimization.

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 ARQ type II codes, flexible rate codes, and static broadcast codes [2], [5]–[7], [9], [11], [12]. This paper focusses on the design of such codes for average power limited additive white Gaussian noise (AWGN) channels.

From a purely information theoretic perspective the problem of rateless transmission is well understood; see, e.g., Shulman [14] 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.

Constructing good codes that also have computationally efficient encoders and decoders requires more effort. A remarkable example of such codes for *erasure* channels are the recent Raptor codes of Shokrollahi [13], which build on the LT codes of Luby [1], [8].

Surprisingly little is known about what is possible beyond the realm of erasure channels. Recent work [4], [10] applies Raptor codes to binary-input AWGN channels (among others), where it is shown that no degree distribution allows Raptor codes to approach capacity simultaneously at different signal to noise ratios (SNRs). Beyond this, binary codes themselves may be "nearly" capacity achieving only at low SNR.

Here, we show that the successful techniques employed to construct low-complexity codes for the standard AWGN channel can be leveraged to construct rateless codes. Specifically, a single codebook designed to operate at a single SNR can be used in a straightforward manner to build a rateless codebook that operates at many SNRs. If the original codebook is good then the rateless codebook will also be good.

We use three tools to construct the rateless codebook:

- multiplicative dithering,
- layered (superposition) coding, layered via a carefully selected linear transformation G, and
- successive decoding with minimum mean-squared error (MMSE) combining.

Dithering scrambles a single base codebook into a set of L quasi-independent codebooks. Layering is used to create incremental redundancy blocks from the L codes, where each block is formed using a different L-fold linear combination. Successive decoding with MMSE combining is a computationally attractive and information-lossless alternative to jointly decoding the layers.

Our main result is that the linear transformation G can be selected so that the resulting code is either exactly or very nearly capacity-achieving over a practically useful range of operating conditions, while maintaining a structure suitable for successive decoding.

II. RATELESS CODES

The codes we construct are designed for a complex AWGN channel

$$\boldsymbol{y}_m = \alpha\boldsymbol{x}_m + \boldsymbol{z}_m, \tag{1}$$

$m = 1, 2, \ldots$, where α is a channel gain that varies from receiver to receiver, $\boldsymbol{x}_m$ is a vector of of N input symbols, $\boldsymbol{z}_m$ is a noise vector of N of i.i.d. complex Gaussians, each of variance σ^2, and $\boldsymbol{y}_m$ is the vector of channel output symbols. The channel input is average power limited to power P per symbol. The channel gain α and noise variance σ^2 are assumed known at the receiver but not necessarily at the transmitter.

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) $\boldsymbol{x}_1(w), \boldsymbol{x}_2(w), \ldots$. The receiver accumulates sufficiently many received blocks $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots$ to recover w. The channel gain α may be viewed as a variable parameter in the model; more incremental redundancy is needed to recover w when α is small than when α 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.

III. Motivating example

To illustrate the key features of the problem, we construct an incremental redundancy code that employs two layers of coding to support a total of two redundancy blocks. The code is "perfect" in a sense that will be made clear presently.

The channel gain $|\alpha|$ may be divided into three intervals based on the number of blocks of redundancy needed for decoding. When $|\alpha| \geq |\alpha_1|$ decoding requires only one block. When $|\alpha_1| > |\alpha| \geq |\alpha_2|$ decoding requires two blocks. When $|\alpha_2| > |\alpha|$ decoding is not possible. The interesting cases occur when the gain is as small as possible to permit decoding. At these critical points, for one-block decoding the decoder sees $\boldsymbol{y}_1 = \alpha_1 \boldsymbol{x}_1 + \boldsymbol{z}_1$, while for two-block decoding the decoder sees

$$\boldsymbol{y}_1 = \alpha_2 \boldsymbol{x}_1 + \boldsymbol{z}_1, \tag{2}$$
$$\boldsymbol{y}_2 = \alpha_2 \boldsymbol{x}_2 + \boldsymbol{z}_2. \tag{3}$$

Let $\text{SNR}_i = P|\alpha_i|^2/\sigma^2$. The capacity of the one-block channel is

$$I_1 = \log_2(1 + \text{SNR}_1), \tag{4}$$

while for the two-block channel the capacity is

$$I_2 = 2\log_2(1 + \text{SNR}_2) \tag{5}$$

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

Let the transmitted message be NR bits long, where R is a design parameter. Since we deliver the same message to the receiver for both the one- and two-block cases, the smallest values of $|\alpha_1|$ and $|\alpha_2|$ we can hope to achieve occur when $I_1 = I_2 = R$. We say that the code is perfect if it is decodable at these limits.

We introduce layering by requiring the transmitted blocks to be linear combinations of two codewords $\boldsymbol{c}_1 \in \mathcal{C}_1$ and $\boldsymbol{c}_2 \in \mathcal{C}_2$:

$$\boldsymbol{x}_1 = g_{1,1}\boldsymbol{c}_1 + g_{1,2}\boldsymbol{c}_2, \tag{6}$$
$$\boldsymbol{x}_2 = g_{2,1}\boldsymbol{c}_1 + g_{2,2}\boldsymbol{c}_2. \tag{7}$$

Codebook $\mathcal{C}_1$ has rate R_1 and codebook $\mathcal{C}_2$ has rate R_2, where $R_1 + R_2 = R$, so that total rate of the two codebooks equals channel capacity. We assume that the codebooks are (essentially) capacity-achieving for an additive white Gaussian noise channel. For convenience, we scale the codebooks to

have unit power; the power constraint instead enters through the constraints

$$|g_{1,1}|^2 + |g_{1,2}|^2 = P, \tag{8}$$
$$|g_{2,1}|^2 + |g_{2,2}|^2 = P. \tag{9}$$

The decoder first recovers $\boldsymbol{c}_2$ while treating $\boldsymbol{c}_1$ as additive Gaussian noise, then recovers $\boldsymbol{c}_1$ using $\boldsymbol{c}_2$ as side information. Our aim is to select $\{g_{i,j}\}$ to enable such decoding for both the one- and two-block cases. The resulting layered rateless code will then be perfect in the sense introduced above.

Finding the admissible $\{g_{i,j}\}$ is a now a matter of some algebra. In the one-block case we need

$$R_1 = I_{\alpha_1}(C_1; Y_1|C_2) \tag{10}$$
$$R_2 = I_{\alpha_1}(C_2; Y_1) \tag{11}$$

and in the two-block case we need

$$R_1 = I_{\alpha_2}(C_1; Y_1, Y_2|C_2) \tag{12}$$
$$R_2 = I_{\alpha_2}(C_2; Y_1, Y_2). \tag{13}$$

The subscripts α_1 and α_2 are a reminder that these equations depend on the channel gain, and the non-boldface symbols denote single components from the input and output vectors.

The messiest part of the calculation occurs when decoding $\boldsymbol{c}_2$ in the two-block case (i.e., in (13)) because $\boldsymbol{c}_1$ causes the effective noise in the two blocks to be correlated. We can avoid this calculation by observing 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 the matrix $[g_{i,j}]$ are orthogonal. Moreover, the power constraint P ensures that these orthogonal rows have the same length, which implies that $[g_{i,j}]$ 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 to $R/2$. This occurs because the two-block case decomposes into two parallel orthogonal channels of equal SNR.

Using (5) and (10) with $I_2/2 = R_1$, we thus have

$$\log_2(1 + \text{SNR}_2) = \log_2(1 + \text{SNR}_1|g_{1,1}|^2/P), \tag{14}$$

so that

$$|g_{1,1}|^2 = P\frac{\text{SNR}_2}{\text{SNR}_1}. \tag{15}$$

The unitary constraint and the power constraint, together with some algebra, allow us to express the remaining coefficients as

$$\begin{bmatrix} g_{1,1} & g_{1,2} \\ g_{2,1} & g_{2,2} \end{bmatrix} = \sqrt{\frac{P}{2^{R/2}+1}} \begin{bmatrix} 1 & 2^{R/4} \\ 2^{R/4} & -1 \end{bmatrix} \tag{16}$$

where we have exploited the freedom to adjust complex phases to make all quantities real. It is straightforward to verify that (10)–(13) are satisfied with this selection.

Note that in this construction, the codebooks $\mathcal{C}_1$ and $\mathcal{C}_2$ cannot be identical. However, in practice suitable codebook pairs can be readily generated from a single base codebook $\mathcal{C}$. For example, it would be sufficient to apply a pseudorandom $\{+1, -1\}$ dither to a single base codebook $\mathcal{C}_1$ to generate $\mathcal{C}_2$.

This 2-layer 2-block construction is our first example of a perfect rateless code formed by taking linear combinations of codewords drawn from equal-rate codebooks. The code can be decoded one layer at a time with no loss in performance, provided the decoder is cognizant of the correlated noise caused by undecoded layers.

IV. RATELESS CODES VIA LINEAR COMBINING

The general approach we adopt for building a rateless code may be described as follows. First, select the maximum number M of redundancy blocks, together with a base rate R for the one-block version of the code. Next, select the number of layers L and codebooks $\mathcal{C}_l, \ldots, \mathcal{C}_L$ each of rate R/L. As we shall see presently, some performance loss is inevitable when M exceeds L.

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

$$\begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_M \end{bmatrix} = G \begin{bmatrix} \boldsymbol{c}_1 \\ \vdots \\ \boldsymbol{c}_L \end{bmatrix} \tag{17}$$

where G is an $M \times L$ matrix of complex gains and where $\{\boldsymbol{x}_m\}$ and $\{\boldsymbol{c}_l\}$ are row vectors of length N. The power constraint enters by limiting the rows of G to have norm P and by normalizing the codebooks to have power 1.

Our aim is to select G so that capacity is achieved for any number $m = 1, \ldots, M$ of redundancy blocks and to allow layered decoding.

Decoding for layered codes is done successively, first decoding $\boldsymbol{c}_L$ while treating $G[\boldsymbol{c}_1^T \ \ldots \ \boldsymbol{c}_{L-1}^T]^T$ as noise, then decoding $\boldsymbol{c}_{L-1}$, treating $G[\boldsymbol{c}_1^T \ \ldots \ \boldsymbol{c}_{L-1}^T]^T$ as noise, and so on. Note that the undecoded layers act as colored noise. The decoder must take this into account, for example by using a minimum mean-squared error combiner on the received blocks $\{Y_m\}$.

What, then, are the constraints on G needed for successive decoding? Successive decoding is possible only if the mutual information at each layer l and block m is at least as large as the rate of code $\mathcal{C}_l$. We can write these constraints more concretely as follows. Let C_l denote a single symbol of $\boldsymbol{c}_l$ (say the first), let C_{l+1}^L denote the set $C_{l+1}, C_{l+2}, \ldots, C_L$, and let $G_{[m,l]}$ denote the $m \times l$ upper-left submatrix of G. Finally, let the critical channel gains $\{\alpha_m\}$ be defined as in Section III, i.e., as the solution to

$$R/L = m \log_2(1 + P|\alpha_m|^2/\sigma^2). \tag{18}$$

Then, to enable successive decoding we require

$$R/L \leq I(C_l; Y_1, \ldots, Y_m \mid C_{l+1}^L) \tag{19}$$

$$= I(C_l; \alpha_m G_{[m,l]}[C_1 \ldots C_l]^T + [Z_1 \ldots Z_m]^T) \tag{20}$$

$$= \log \frac{\det(\sigma^2 \boldsymbol{I}_m + |\alpha_m|^2 G_{[m,l]} G_{[m,l]}^H)}{\det(\sigma^2 \boldsymbol{I}_m + |\alpha_m|^2 G_{[m,l-1]} G_{[m,l-1]}^H)}, \tag{21}$$

for $l = 1, \ldots, L$ and $m = 1, \ldots, M$, where $\boldsymbol{I}_m$ is the $m \times m$ identity matrix. The power constraint ensures that the

inequalities (21) must be met with equality, and that the overall code meets channel capacity.

Thus, our ability to either exactly or approximately satisfy (21) determines our ability to construct good rateless codes via linear combining (17).

When $L < m$ these constraints cannot be met exactly because an L-fold linear combination cannot fill all degrees of freedom afforded by the m-block channel. Specifically, the capacity of the m-block channel is

$$I_m = m \log(1 + |\alpha|^2 P/\sigma^2) \tag{22}$$

while the rate achievable using a linear combination of L codes is bounded by

$$R_m = \begin{cases} m \log(1 + |\alpha|^2 P/\sigma^2) & \text{for } m \leq L, \\ L \log(1 + |\alpha|^2 (m/L)(P/\sigma^2)) & \text{for } m > L. \end{cases} \tag{23}$$

To meet the upper bound for $m \leq L$ requires that the linear combination of L codebooks create an i.i.d. Gaussian sequence. To meet the upper bound 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). Thus, a necesssary condition to reach the upper bound is that the rows of $G_{[m,L]}$ be orthogonal for $m = L$ and that the columns of $G_{[m,L]}$ be orthogonal for $m \geq L$.

If we fix a target rate R for the rateless code, we can use (22) to find the smallest channel gain α_m for which decoding with m blocks is possible:

$$|\alpha_m|^2 = (e^{R/m} - 1)\frac{\sigma^2}{P}. \tag{24}$$

If we instead use the bound (23) that applies to rateless codes constructed using linear combinations of L base codes, we find

$$|\alpha_m'|^2 = \begin{cases} (e^{R/m} - 1)\frac{\sigma^2}{P} & \text{for } m \leq L, \\ (e^{R/L} - 1)(L/m)\frac{\sigma^2}{P} & \text{for } m > L. \end{cases} \tag{25}$$

The performance loss $20 \log_{10} |\alpha_m'|/|\alpha_m|$ caused by the linear structure is shown in Table I for $R = 5$ bits per complex symbol. If an application requires $M = 10$ redundancy blocks, for example, a three-layer code has a worst-case loss of less than 2 dB, while a five-layer code has a worst-case loss of less than 0.82 dB.

When $M = L$ the bounds (23) present no barrier to reaching channel capacity. Yet the conditions under which (21) may be satisfied are not obvious. We have already seen one example (16) with $L = M = 2$. Routine if lengthy algebra yield others. A construction that meets (21) when $L = M = 3$ and $R = 6$ bits per complex symbol is as follows:

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

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

	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

TABLE I

LOSSES $|\alpha'_m|/|\alpha_m|$ IN dB CAUSED BY LINEAR STRUCTURE FOR A RATELESS CODE OF BASE RATE $R = 5$ BITS/SYMBOL, AS A FUNCTION OF THE NUMBER OF LAYERS L AND THE NUMBER OF REDUNDANCY BLOCKS.

where

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

$$\theta_3 = -\arctan \sqrt{7} \quad , \quad \theta_4 = \pi - \arctan \sqrt{7}/3$$

The rate $R = 6$ was selected because the resulting expressions are unusually compact; examples for other rates may be readily found.

For $M > 3$ the algebra becomes daunting, though we conjecture that exact solutions and hence perfect rateless codes exist for all $L = M$. We instead tackle the problem using numerical optimization, as developed in the next section.

V. NUMERICAL EXAMPLES

We have experimented with numerical optimization methods to satisfy (21) for up to $M = 10$ redundancy blocks, using the critical channel gains α'_m in place of α_m as appropriate when the number of blocks M exceeds the number of layers L.

In all cases $M = 2, \ldots, 10$, we found constructions with $L = M$ and $R/L = 2$ bits per complex symbol that come within one part in 1000 of satisfying (21) with equality, and often the solutions come within one part in 10000. This provides powerful evidence that perfect rateless codes exist for a wide range of parameter choices.

For $L < M$ perfect constructions cannot exist, as developed in the previous section. Moreover, it is not even possible to meet (23) with equality. That would entail the construction of orthogonal m-dimensional vectors (with nonzero entries) that remain orthogonal when truncated to their first $m - 1$ dimensions, an obvious impossibility.

Despite this, in most cases of interest one can come remarkably close to satisfying (21) (as modified using the channel gains α'_m from (25)). 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 shortfall in meeting the mutual information constraints (21) for a 3-layer code with

	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

TABLE II

PERCENT SHORTFALL IN RATE FOR A NUMERICALLY-OPTIMIZED RATELESS CODE WITH $M = 10$ BLOCKS, $L = 3$ LAYERS, AND A BASE RATE OF $R = 5$ BITS PER COMPLEX SYMBOL.

$M = 10$, and $R = 5$, whose complex gain matrix is

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

The worst case loss is less than 1.5%. Note that this loss is cumulative with the loss in Table I.

The example in Table II is typical in its efficiency. As a practical matter, the performance loss caused by the linear layered structure we have imposed on the rateless code arises almost entirely from the code's inability to occupy all available degrees of freedom in the channel, and can therefore be estimated quite closely using (22) and (23), as in Table I.

VI. INCREASING BREAKPOINT RESOLUTION

With an ideal rateless code, *every* prefix of the code is a capacity-achieving code. This corresponds to a particularly dense set of SNR breakpoints at which decoding can occur. By contrast, the rateless codes we develop in the preceding sections have the property that only prefixes whose lengths are an integer multiple of the base block length are both capacity-achieving and efficiently decodable. This sparseness of SNR breakpoints can be undesirable in some applications, since when the realized SNR is between breakpoints, capacity is no longer achieved: the realized rate is that corresponding to the next lower SNR breakpoint.

One approach for controlling this aspect of our rateless code behavior is as follows. Suppose we are interested in a rateless code whose base rate (i.e., the maximum rate at which the code is capacity-achieving) is R. Then we use the rateless construction of the preceding section to design a code of base rate is κR, where $1 \le \kappa \le M$, and have the decoder collect at least κ blocks before attempting to decode. With this approach, the associated rate breakpoints are $R, R\kappa/(\kappa + 1), R\kappa/(\kappa + 2), \ldots, R\kappa/M$, where we note that the largest rate increment is the first, corresponding to the factor $\kappa/(\kappa + 1)$. Hence, by choose larger values of κ, one can increase the density of rate (and thus SNR) breakpoints.

It should be stressed, however, that there is a price to paid with this approach. In particular, if we keep constant the number of codeword symbols that must be accumulated before decoding at rate R is possible, then the underlying block size in our rateless construction must decrease inversely with κ. Thus, for sufficiently large κ the basic block length becomes short enough that code performance suffers, and so in practice the selection of κ involves a compromise.

VII. Existence of Nearly-Rateless Codes

The construction of perfect rateless codes becomes more challenging with increasing M. In this section we show that, by contrast, it is easy to construct rateless codes for any M that are arbitrarily close to perfect in an appropriate sense, provided enough layers are used. We term these nearly-rateless codes.

Our nearly-rateless codes [3] slightly generalize the code construction in (17) to allow the combining matrix G to vary from symbol to symbol. In particular, given codewords $c_l \in \mathcal{C}_l$, $l = 1, \dots, L$, the redundancy blocks $x_1, \dots, x_M$ now have the form

$$
\begin{bmatrix} x_1(n) \\ \vdots \\ x_M(n) \end{bmatrix} = G(n) \begin{bmatrix} c_1(n) \\ \vdots \\ c_L(n) \end{bmatrix}, \tag{26}
$$

where $n = 1, \dots, N$. This time-variation simplifies the analysis by allowing simple averaging arguments to be exploited. The power constraint enters as before by limiting the rows of $G(n)$ to have a squared norm P and by normalizing the codebooks to have unit-energy codewords.

It suffices to restrict $G(n)$ to be of the form

$$
G(n) = P \odot D(n),
$$

where P is an $M \times L$ power allocation matrix with extries $\sqrt{P_{m,l}}$ that do not vary within a block, and where $D(n)$ is a phase-only "dither" matrix. with $\odot$ denoting componentwise multiplications.

By (21), the mutual information of the lth layer between the channel input and output for a channel gain α_m upon receiving m blocks is

$$
I_{l,m} = \frac{1}{N} \sum_{n=1}^{N} I_{l,m}(n)
$$

where

$$
I_{l,m}(n) = \log \frac{\det(\sigma^2 \boldsymbol{I}_m + |\alpha_m|^2 G_{[m,l]}(n) G_{[m,l]}(n)^H)}{\det(\sigma^2 \boldsymbol{I}_m + |\alpha_m|^2 G_{[m,l-1]}(n) G_{[m,l-1]}(n)^H)}.
$$

Then if the entries of $D(n)$ are random phases drawn independently for each $n = 1, \dots, N$, the law of large numbers implies that one can achieve a per-layer rate of

$$
\lim_{N \to \infty} I_{l,m} = E_D\left[I_{l,m}(1)\right] = \bar{I}_{l,m}. \tag{27}
$$

When the efficiency of such rateless code is defined as

$$
\eta = \frac{\min_{l,m} \bar{I}_{l,m}}{R/L}
$$

it can be verified [3] that the efficiency can be made arbitrarily close to unity by taking sufficiently many layers L and

choosing the power matrix P according to the following recursion:

$$
\begin{aligned}
P_{m+1,l} = &(2^{\Delta_{m+1,l}} - 1) \\
&\cdot (P_{m+1,1} + \dots + P_{m+1,l-1} + \frac{\sigma^2}{|\alpha_{m+1}|^2}), \quad (28)
\end{aligned}
$$

where

$$
\Delta_{m+1,l} = \frac{R}{L} - \sum_{m'=1}^{M} \log(1 + \mathrm{SNR}_{m',l}(\alpha_{m+1})), \tag{29}
$$

with

$$
\mathrm{SNR}_{m,l}(\alpha) = \frac{|\alpha|^2 P_{m,l}}{|\alpha|^2 (P_{m,1} + \dots + P_{m,l-1}) + \sigma^2}. \tag{30}
$$

The recursion proceeds from $m = 1$ to $m = M$ and for each m goes over $l = 1, \dots L$.

With this construction, the per-layer channel is now a time-varying one, which is unattractive from the perspective of base code design and potentially decoding complexity. However, both these issues are addressed by exploiting maximal-ratio combining in conjunction with successive-cancellation decoding at the receiver, which transforms the per-layer channel back into a time-invariant one. Thus, this change allows any of the many capacity-approaching codes with low complexity decoding algorithms to be used as the base code in the design.

References

[1] J. W. Byers, M. Luby, and M. Mitzenmacher, "A Digital Fountain Approach to Asynchronous Reliable Multicast," *IEEE J. Select. Areas Commun.*, vol. 20, pp. 1528–1540, Oct. 2002.

[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] U. Erez, G. W. Wornell, and M. Trott, "Coding for Faster-than-Nyquist signaling: the merits of a regime change," *42nd Annual Allerton Conference on Communication, Control, and Computing*, Monticello, Illinois, Sept. 29–Oct. 1, 2004.

[4] O. Etesami, M. Molkaraie, and A. Shokrollahi, "Rateless codes on symmetric channels," *Int. Symp. Inform. Theory (ISIT2004)*, Chicago, Illinois, June 2004, p. 38.

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

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

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

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

[9] D. M. Mandelbaum, "An adaptive feed-back coding using incremental redundancy," *IEEE Trans. Inform. Theory*, vol. 20, pp. 388–389, May 1974.

[10] R. Palanki and J.S. Yedidia, "Rateless codes on noisy channels," *Int. Symp. Inform. Theory*, Chicago, Illinois, June 2004, p. 37.

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

[12] 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.

[13] A. Shokrollahi, "Raptor codes," (preprint, 2002).

[14] N. Shulman, *Universal channel coding*, Ph.D. thesis, Tel-Aviv University, 2004.