I. INTRODUCTION

In a communication system, it is often the case that the channel is not constant. The transmitter may not be aware of the state of the channel, but for the decoder it is in general not very difficult to find out what the actual state is (was). Certain fading channels correspond to such a situation. Note that the channel can be changing (fading) over time and/or over frequency. Despite the fact that the transmitter is not aware of the state of the channel it would be desirable if the largest possible rate could be achieved at any time and/or for all frequencies. Here we want to investigate how this can be realized. We therefore model the varying channel as a collection of parallel channels and study coding techniques for this situation. We first focus on the case where the same input distribution achieves capacity for all channels in the collection but later we will also consider the case where this assumption does not hold.

The outline of the correspondence is as follows. In Section II, we describe the model that we use in our investigations. Essential is the concept of an input-permuter that connects code-sequences in an arbitrary way to each of the parallel channels. The permutation remains constant during the (simultaneous) transmission of the code-sequences via the parallel channels. The transmitter does not know how the sequences are permuted. The permuted parallel channels is studied here. The transmitter does not know the state of the channel. We first consider the case where all channels in the collection have identical varying channel as a collection of parallel channels and study coding techniques for this situation. We focus on the case where the same input distribution achieves capacity for all channels in the collection but later we will also consider the case where this assumption does not hold.

The problem of transmission of information over arbitrarily permuted parallel channels is studied here. The transmitter does not know over which channel a certain code-sequence will actually be transmitted, however the receiver knows how the sequences are permuted. The permutation is arbitrary but constant during the (simultaneous) transmission of the code-sequences via the parallel channels. It is shown first that the sum of the capacities of each channel is achievable for such a communication system in the special case where the capacity achieving input distributions of all channels are identical. More important is that this sum-capacity can also be achieved using a single channel code for all channels combined with a sequential decoding method. The construction of a rate-matching code based on Maximum Distance Separable (MDS) codes turns out to be crucial. Finally, the case where the parallel channels have different capacity-achieving input distributions is investigated. Also for this case the capacity is determined. Again, this capacity is achievable with a sequential decoding procedure.

Index Terms—Capacity, maximum distance separable (MDS) codes, parallel channels, permuted channels, rate-matching code, sequential decoding method.

REFERENCES


Signalizing Over Arbitrarily Permuted Parallel Channels

Frans M. J. Willems, Fellow, IEEE, and Alexei Gorokhov, Member, IEEE

Abstract—The problem of transmission of information over arbitrarily permuted parallel channels is studied here. The transmitter does not know over which channel a certain code-sequence will actually be transmitted, however the receiver knows how the sequences are permuted. The permutation is arbitrary but constant during the (simultaneous) transmission of the code-sequences via the parallel channels. It is shown first that the sum of the capacities of each channel is achievable for such a communication system in the special case where the capacity achieving input distributions of all channels are identical. More important is that this sum-capacity can also be achieved using a single channel code for all channels combined with a sequential decoding method. The construction of a rate-matching code based on Maximum Distance Separable (MDS) codes turns out to be crucial. Finally, the case where the parallel channels have different capacity-achieving input distributions is investigated. Also for this case the capacity is determined. Again, this capacity is achievable with a sequential decoding procedure.

Index Terms—Capacity, maximum distance separable (MDS) codes, parallel channels, permuted channels, rate-matching code, sequential decoding method.

I. INTRODUCTION

In a communication system, it is often the case that the channel is not constant. The transmitter may not be aware of the state of the channel, but for the decoder it is in general not very difficult to find out what the actual state is (was). Certain fading channels correspond to such a situation. Note that the channel can be changing (fading) over time and/or over frequency. Despite the fact that the transmitter is not aware of the state of the channel it would be desirable if the largest possible rate could be achieved at any time and/or for all frequencies. Here we want to investigate how this can be realized. We therefore model the varying channel as a collection of parallel channels and study coding techniques for this situation. We first focus on the case where the same input distribution achieves capacity for all channels in the collection but later we will also consider the case where this assumption does not hold.

The outline of the correspondence is as follows. In Section II, we describe the model that we use in our investigations. Essential is the concept of an input-permuter that connects code-sequences in an arbitrary way to each of the parallel channels. The permutation remains constant as long as it takes to transmit the code-sequences via the channels. The permuter embodies the fact that the transmitter does not know the actual state of the channel. We first consider the case where all channels have the same capacity-achieving input distribution. Section III shows...
that the capacity of the system is the sum of the capacities of the parallel channels. The proof of this statement is based on the asymptotic equipartition property (AEP), see Cover and Thomas [1]. In Section IV, we propose a sequential decoding procedure combined with a single code for all channels. A crucial ingredient of this method is a so-called rate-matching code that creates the required dependency between the codewords for the parallel channels. We define the capacities of a rate-matching code and discuss some of its properties. Then in Section V we use the AEP to demonstrate that sequential procedures achieve the sum-capacity. In Section VI we propose several methods that can be used to construct rate-matching codes. Section VII discusses an application of the results that we have obtained here. In particular we show that for two flat-fading additive white Gaussian noise (AWGN) channels in parallel, we can decrease the outage probability. The case where the parallel channels have different capacity-achieving input distributions is considered in Section VIII. We determine the capacity also for this case. We conclude by making some final remarks in Section IX.

II. PROBLEM DESCRIPTION

Suppose that we have a communication system in which the transmitter (encoder) is connected to the receiver (decoder) by S parallel channels (see Fig. 1 where S = 3). The encoder produces S code-sequences, all having length T. These S code-sequences are transmitted over the S channels, each sequence via one of them. Within T transmissions a message-index W is conveyed by the encoder to the decoder. This index assumes values in {1, 2, . . . , M}. The distribution of W is uniform, i.e., Pr{W = w} = 1/M for all w ∈ {1, 2, . . . , M}.

The input alphabets of all S parallel channels are assumed to be discrete and identical; thus

\[ X_1 = X_2 = \cdots = X_S = X. \]  

(1)

For each message-index W the encoder generates S code-sequences \( X_1, X_2, \ldots, X_S \), hence

\[ (X_1, X_2, \ldots, X_S) \equiv \epsilon(W) \]  

(2)

where \( X_s \equiv (x_{s1}, x_{s2}, \ldots, x_{sT}) \) for \( s = 1, \ldots, S \), with \( x_{st} \in X \) for \( t = 1, 2, \ldots, T \).

An input-permuter then permutes the S code-sequences over the S parallel channels. More specifically it “connects” channel \( s = 1, \ldots, S \) hence code-sequence \( X_s \) is transmitted via channel \( s = \pi^{-1}(r) \). The mapping \( \pi(\cdot) \) from \( \{1, \ldots, S\} \) to \( \{1, \ldots, S\} \) is one-to-one (a permutation). Since there are \( S! \) different permutations we label these permutations 1, 2, . . . , S! here. At a certain moment prior to transmission, one of these permutations is chosen. This permutation then remains constant as long as it takes to transmit the S code-sequences simultaneously via the parallel channels. The encoder is not aware of the chosen permutation \( \pi \), the decoder is supposed to be informed about \( \pi \) however (or is able to determine it somehow).

All S parallel channels are memoryless. Channel \( s \) for \( s = 1, \ldots, S \) has transition probability matrix \( \{P_s(x_s|x), x \in X, y_s \in Y_s\} \). Note that the discrete channel output alphabets \( Y_1, Y_2, \ldots, Y_S \) need not be identical.

The decoder, knowing the actual permutation \( \pi \), forms an estimate \( \hat{W}(\pi) \in \{1, 2, \ldots, M\} \) of the transmitted message-index based on the received S channel output sequences \( Y_{1s}, Y_{2s}, \ldots, Y_{Ss} \), i.e.

\[ \hat{W}(\pi) = d(Y_{1s}, Y_{2s}, \ldots, Y_{Ss}, \pi). \]  

(3)

Here channel output sequence \( Y_{ss} \), for \( s = 1, \ldots, S \), is the output of channel \( s \), hence

\[ Y_{ss} \equiv (Y_{s1}, Y_{s2}, \ldots, Y_{sT}) \]  

(4)

for \( Y_{st} \in Y_s \) for \( t = 1, 2, \ldots, T \).

We now say that rate \( R \) is achievable if, for all \( \delta > 0 \) and all \( T \) large enough, there exist an encoder \( \epsilon(\cdot) \) and decoder \( d(\cdot, \cdot) \) such that both

\[ \frac{1}{T} \log_2 M \geq R - \delta \]

\[ \Pr \{\hat{W}(\pi) \neq W\} \leq \delta, \]  

(5)

for all \( S! \) permutations \( \pi \).

The largest achievable rate is called the capacity \( C_H \). In the next section we will determine this capacity for the case that one single input distribution achieves capacity for all S parallel channels.

III. BASIC RESULT

The capacity of channel \( s \) for \( s = 1, \ldots, S \) in bits per transmission is

\[ C_s = \max_{Q(\cdot)} I(X; Y_s) \]  

(6)

for

\[ I(X; Y_s) = \sum_{x \in X} Q(x) P_s(y_s|x) \log_2 \frac{P_s(y_s|x)}{\sum_{x' \in X} Q(x') P_s(y_s|x')} \]  

(7)

where in the summations \( x \) and \( x' \) run over \( X \) and \( y_s \) over \( Y_s \). Moreover \{\( Q(x), x \in X\}\} is the channel input distribution.

Theorem 1: The capacity for transmission over S arbitrarily permuted parallel channels equals

\[ C_H = \sum_{s=1}^{S} C_s, \]  

(8)

when all channels have an identical input alphabet \( X \) and the same capacity-achieving input distribution \( \{Q^*(x), x \in X\} \).

Note that a certain channel may not have a unique capacity-achieving input distribution. What matters in the theorem however is that there is an input distribution \( Q^*(\cdot) \) that achieves capacity for all parallel channels.

In what follows we will prove this result. First note that \( \sum_{s=1}^{S} C_s \) is the capacity if both the encoder and the decoder know the actual permutation \( \pi \). Since the encoder is not aware of the actual permutation we obtain the trivial upper bound

\[ C_H \leq \sum_{s=1}^{S} C_s. \]  

(9)

This upper bound turns out to be achievable as we will see next. Observe that for each of the \( S! \) permutations \( \pi \) the channel between encoder and decoder is a memoryless product-channel with input alphabet


\[ P_r \left( \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_S \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_S \end{pmatrix} \right) = \Pi_{i=1}^S Q_i(x_{i}) \]

where \( Q_i(\cdot) \) is the distribution achieving capacity for all \( S \) channels. Then for all permutations \( \pi \)

\[ I(X_1, X_2, \ldots, X_S; Y_1, Y_2, \ldots, Y_S) = \sum_{i=1}^S I(X_{\pi(i)}; Y_i) \]

\[ = \sum_{i=1}^S C_i. \]

We now design a code whose code-blocks \( X_0 \triangleq \{ X_s, s = 1, \ldots, S, t = 1, \ldots, T \} \) are transmitted over all these \( S! \) product channels simultaneously, see Fig. 2, and for which the error probabilities realized by all \( S! \) decoders connected to these channels, can be made arbitrarily small. Note that \( P_r \{ \hat{W}(\pi) \neq W \} \) denotes the error probability for the decoder that corresponds to permutation \( \pi \).

Consider a random coding argument along the lines of Cover and Thomas [1, Ch. 7]. Generate for each message index \( w \in \{ 1, 2, \ldots, M = 2^{S^T} \} \) an \( S \times T \) code-block \( X_0(w) \triangleq \{ X_s, s = 1, \ldots, S, t = 1, \ldots, T \} \), where each component of this block gets value \( x \in X \) with probability \( Q_i(x) \) independently of all other components. Note that this code is a random code for all \( S^T \) product channels where the input distribution of a column (super-input) is given by (10).

Every decoder \( \pi \) can now perform decoding by joint typicality, it has to search for a message index \( w \) such that the corresponding code-block \( X_0(w) \) is jointly typical with the received block \( Y_0 \triangleq \{ Y_s, s = 1, \ldots, S, t = 1, \ldots, T \} \), relative to the super-channel determined by \( \pi \). By (11), for rate \( R \triangleq \sum_{i=1}^S C_i - 4 \epsilon \), the error probability averaged over the ensemble of codes \( P_r \{ \hat{W}(\pi) \neq W \} \) is not larger than \( 2 \epsilon \) for each decoder \( \pi \), for all large enough block lengths \( T \). Here \( \epsilon > 0 \) is the parameter that specifies the typical set \( A^T_\epsilon \). The probability (averaged over the ensemble of codes) \( P_r \{ \hat{W}(\pi) \neq W \} \) that any of these \( S! \) decoders produces the wrong estimate is therefore not larger than \( 2S! \epsilon \). Consequently there exists at least one code with total error probability \( P_r \{ \hat{W}(\pi) \neq W \} \) not larger than \( 2S! \epsilon \). Therefore, for this code, for all \( S! \) decoders \( \pi \in \{ 1, 2, \ldots, S! \} \) the error probability

\[ P_r \{ \hat{W}(\pi) \neq W \} \leq 2S! \epsilon. \]

If we let \( \epsilon \to 0 \), we can make this error probability and the difference between \( R \) and \( \sum_{i=1}^S C_i \) arbitrarily small. Therefore we may conclude that \( \sum_{i=1}^S C_i \) is achievable. Together with the upper bound (9) this proves Theorem 1.

Note that our achievability proof resembles the proof given by Cover, McEliece, and Posner [2] for the asynchronous multiple-access channel. In [2] a single code has to be good for all possible delays, here one code must be reliable for all possible permutations.

### IV. A Sequential Decoding Procedure

The product-channel approach that was considered in the previous section leads to a large decoding complexity. The decoder has to check all code-blocks \( X_0(w) \) for \( w = 1, \ldots, 2^{S^T} \) against the received block \( Y_0 \triangleq \{ Y_s, s = 1, \ldots, S, t = 1, \ldots, T \} \). In this section we show that \( \sum_{i=1}^S C_i \) can also be achieved using a single code for all channels and performing a sequential decoding procedure. To make things simple we focus on the case \( S = 3 \) here. The method that we describe generalizes to larger \( S \) however. Also the case where \( S = 2 \) will be discussed.

First, we will describe an important ingredient of a sequential decoding procedure, i.e., the so-called rate-matching code. In Section V we will then prove that it is possible to achieve capacity with a sequential procedure.

To make the decoding effort simpler we use a single code for each channel. Assume that this channel code consist of \( M_i = 2^{TR_i} \) codewords \( \{ 1 \}, \{ 2 \}, \ldots, \{ M_i \} \). Codeword \( x(wi) = (x_1(w_i), x_2(w_i), \ldots, x_T(w_i)) \) is a sequence of \( T \) symbols from \( X \), for \( w_i = 1, \ldots, M_i \).

It is obvious that we need some dependency between the codewords that are transmitted over the three channels. This dependence is created by the rate-matching code. This code maps the message-index \( w \) onto a triple \( (w_1, w_2, w_3) \) of indices. These three indices specify the codewords \( x(w_1), x(w_2) \), and \( x(w_3) \) that are then sent over three arbitrarily permuted channels. A rate-matching code for \( S = 3 \) is therefore defined by three mappings

\[ f_1: \{ 1, 2, \ldots, M \} \to \{ 1, 2, \ldots, M_1 \} \]
\[ f_2: \{ 1, 2, \ldots, M \} \to \{ 1, 2, \ldots, M_2 \} \]
\[ f_3: \{ 1, 2, \ldots, M \} \to \{ 1, 2, \ldots, M_3 \}. \]

These mappings result in three kinds of subsets of \( \{ 1, 2, 3, \ldots, M \} \). For \( i, j, k \in \{ 1, 2, 3 \} \) and \( i \neq j, j \neq k, k \neq i \) we define the subsets

\[ B_{i,j} \triangleq \{ f_i(w) : \text{for } w = 1, 2, \ldots, M \} \]
\[ B_{i,j,k} \triangleq \{ f_i(w) : \text{for } w = 1, 2, \ldots, M \text{ such that } f_j(w) = w_j \}
\[ B_{i,j,k} \triangleq \{ f_i(w) : \text{for } w = 1, 2, \ldots, M \text{ such that } f_j(w) = w_j \text{ and } f_k(w) = w_k \}. \]

We are now ready to state what we mean by a rate-matching code.

**Definition 1:** An \( (M, M_1, M_2, M_3) \)-rate-matching code must be one-to-one, hence for all \( w = 1, 2, \ldots, M, w' = 1, 2, \ldots, M \), and \( w' \neq w \)

\[ (f_1(w), f_2(w), f_3(w)) \neq (f_1(w'), f_2(w'), f_3(w')). \]

Moreover an \( (M, M_1, M_2, M_3) \)-rate-matching code must satisfy for all \( i, j, k \in \{ 1, 2, 3 \} \) and \( i \neq j, j \neq k, k \neq i \) the three equalities

\[ |B_{i,j}| = M_i \]
\[ |B_{i,j}(w_j)| = M_2 \]
\[ |B_{i,j,k}(w_j, w_k)| = M_3 \]

for all \( w_j \in \{ 1, 2, \ldots, M_1 \} \) and \( w_k \in B_{i,j}(w_j) \).

A first consequence of (14) and Definition 1 is that for an \( (M, M_1, M_2, M_3) \)-rate-matching code

\[ M_1 \geq M_2 \geq M_3. \]
Finally, from the actual permutation \(\pi\) possible indices \(\bar{w}_{\pi(1)}\) of the second-best channel, i.e., channel 2. Note that \(\bar{w}_{\pi(1)}\) is the actual permutation. We define \(\bar{w}_{\pi(1)}\) as the inputs to channel 1, 2, and 3, respectively, see Fig. 3. Moreover, the channel with capacity \(C_1\) is the largest capacity, i.e., channel 1. If \(\bar{w}_{\pi(1)}\) is known to the decoder, and that there are only \(M_2\) possible indices \(w_{\pi(2)}\) in \(\mathcal{E}_{2\{1\}}(w_{\pi(1)})\) for the second codeword that need to be considered. When \(\frac{1}{\log_2 M_2}\) is smaller than \(C_2\) reliable reconstruction of \(w_{\pi(2)}\) is feasible. Finally, the output \(Y_3\) of the worst channel, i.e., channel 3, is processed. There are \(M_3\) possible indices \(w_{\pi(3)}\) in \(\mathcal{E}_{3\{1,2\}}(w_{\pi(1)}, w_{\pi(2)})\) that need to be checked now. For \(\frac{1}{\log_2 M_3}\) smaller than \(C_3\) the correct index \(w_{\pi(3)}\) can be found with probability arbitrarily close to one. Finally, from the actual permutation \(\pi\) and \((w_{\pi(1)}, w_{\pi(2)}, w_{\pi(3)})\) the index-triple \((w_1, w_2, w_3)\) and the message-index \(w\) can be determined.

V. ACHIEVABLE PROOF FOR A SEQUENTIAL PROCEDURE

In this section, we will demonstrate the achievability of \(C_1 + C_2 + C_3\) using a single channel code, a rate-matching code, and a sequential decoding procedure.

A. Code Generation

Consider the input distribution \(\{Q^T(x) : x \in \mathcal{X}\}\) that is capacity-achieving for all three channels. We use this distribution to generate a code with \(M_1 = 2^{T R_1}\) codewords \(g(1), g(2), \ldots, g(M_1)\) all having length \(T\) at random. More precisely

\[
\text{Pr}\{\bar{X}(w_1) = g\} = \prod_{i=1}^{T} Q^T(x_i) \quad \text{for all } w_1 = 1, \ldots, M_1. \tag{20}
\]

Here \(x = (x_1, x_2, \ldots, x_T)\). The probability of producing a particular code \(C\) with codewords \(g(1), g(2), \ldots, g(M_1)\) is therefore \(\text{Pr}[C] = \Pi_{w=1}^{M_1} \text{Pr}[\bar{X}(w_1) = g(w_1)]\).

B. Encoding and Transmission

We will use a \((2^{T R_2}, 2^{T R_3})\)-rate-matching code with

\[
R = R_1 + R_2 + R_3 \tag{21}
\]

for \(R_1 \geq R_2 \geq R_3\) that will be specified later. This rate-matching code transforms message index \(w\) into an index-triple \((w_1, w_2, w_3)\). The resulting codewords \(\bar{X}(w_1), \bar{X}(w_2), \) and \(\bar{X}(w_3)\) are now ready for transmission over the three channels, i.e.,

\[
\bar{X}_1 = \bar{X}(w_1), \quad \bar{X}_2 = \bar{X}(w_2), \quad \bar{X}_3 = \bar{X}(w_3). \tag{22}
\]

The input-permuter permutes these codewords over the three channels. For \(s = 1, 2, 3\) the input of channel \(s\) is \(\bar{X}(w_{\pi(s)})\). The decoder receives the three sequences \(Y_1, Y_2,\) and \(Y_3\). Note that

\[
\text{Pr}\{\bar{X}(w_1), \bar{X}(w_2), \bar{X}(w_3)\} = \Pi_{s=1}^{3} \Pi_{w=1}^{M_s} P_s(x_s | (w_{\pi(s)})). \tag{23}
\]

Here \(x_s\) is the \(t\)-th component of codeword \(\bar{X}(w_s)\), for \(w_s = 1, M_s\) and \(t = 1, 2, 3\).

C. Decoding Procedure

The decoder uses a sequential procedure based on decoding by joint typicality. For each definition of the sets of jointly typical sequences \(\mathcal{A}^T(X, Y_1), \mathcal{A}^T(X, Y_2),\) and \(\mathcal{A}^T(X, Y_3)\), we refer to Cover and Thomas [1, Sec. 7.6]. The distributions that determine these sets are \(P(x, y_s) = Q^S(x) P_s(y_s | x)\) for \(s = 1, 2, 3\). Note that by (19) we have

\[
I(X_{\pi(1)}; Y_1) = C_1 \geq I(X_{\pi(2)}; Y_2) \geq C_2 \geq I(X_{\pi(3)}; Y_3) = C_3. \tag{24}
\]

The decoder first decodes the message-index transmitted over the channel with capacity \(C_1\). It declares that the index \(\bar{w}_{\pi(1)}\) was sent if there is a unique index \(\bar{w}_{\pi(1)}\) such that the pair \((\bar{X}(\bar{w}_{\pi(1)}), \bar{X}_1) \in \mathcal{A}_1(X, Y_1)\), i.e., if this pair is jointly typical. If no such \(\bar{w}_{\pi(1)}\) exists or there are more than one such, then an error is declared and decoding stops. If not, the decoder proceeds with the channel having capacity \(C_2\). It declares that \(\bar{w}_{\pi(2)}\) was transmitted if there is a unique index \(\bar{w}_{\pi(2)}\) such that the pair \((\bar{X}(\bar{w}_{\pi(2)}), \bar{X}_2) \in \mathcal{A}_2(X, Y_2)\) and \(\bar{w}_{\pi(2)}\) in \(\mathcal{E}_{2\{1\}}(\bar{w}_{\pi(1)})\). If no such \(\bar{w}_{\pi(2)}\) exists or there are more than one such, an error is declared and decoding stops. If not, the decoder processes the output of channel 3, the channel with capacity \(C_3\). It declares that index \(\bar{w}_{\pi(3)}\) was sent if there is a unique index \(\bar{w}_{\pi(3)}\) such that the pair \((\bar{X}(\bar{w}_{\pi(3)}), \bar{X}_3) \in \mathcal{A}_3(X, Y_3)\) and \(\bar{w}_{\pi(3)}\) in \(\mathcal{E}_{3\{1,2\}}(\bar{w}_{\pi(1)}, \bar{w}_{\pi(2)})\). If there is no such \(\bar{w}_{\pi(3)}\) or there are more than one such, an error is declared.

From the actual permutation \(\pi\) and \((\bar{w}_{\pi(1)}, \bar{w}_{\pi(2)}, \bar{w}_{\pi(3)})\) the index-triple \((w_1, w_2, w_3)\) and the message-index \(w\) can be determined.

D. Analysis of Probability of Error

First we fix a certain permutation \(\pi\) and investigate what happens when \(\pi\) is the actual permutation. We define the following events:

\[
E_{w_1}^1 = \{(\bar{X}(w_1), Y_1) \in \mathcal{A}_1(X, Y_1)\}, \quad E_{w_2}^2 = \{(\bar{X}(w_2), Y_2) \in \mathcal{A}_2(X, Y_2)\}, \quad E_{w_3}^3 = \{(\bar{X}(w_3), Y_3) \in \mathcal{A}_3(X, Y_3)\} \tag{25}
\]

for \(w_1, w_2, w_3 \in \{1, 2, \ldots, 2^{T R_1}\}\). Recall that \(Y_1, Y_2,\) and \(Y_3\) are the results of sending \(W\). Moreover \(W_1, W_2,\) and \(W_3\) are the random
variables induced by the random variable $W$. The error probability averaged over the ensemble of codes for actual permutation $\pi$ is

$$
\Pr\{\hat{W}(\pi) \neq W\} = \Pr\left\{E_{W^{1}}^{1} + \bigcup_{w_{2}=1}^{\varepsilon^{2}} \Pr\{E_{w_{1}}^{1}\} + \bigcup_{w_{3}=1}^{\varepsilon^{3}} \Pr\{E_{w_{2}}^{2}\} + \bigcup_{w_{3}=1}^{\varepsilon^{3}} \Pr\{E_{w_{3}}^{3}\} \right\}
$$

where it is understood that probability \(\Pr\{E_{W^{1}}^{1}\}\) denotes \(\sum_{w_{1}=1}^{\varepsilon^{1}} \Pr\{W = w_{1}\} \Pr\{E_{w_{1}}^{1}\} = \sum_{w_{1}=1}^{\varepsilon^{1}} \Pr\{E_{w_{1}}^{1}\} / \Pr\{E_{W^{1}}^{1}\}\), and the sum of probabilities \(\sum_{w_{3}=1}^{\varepsilon^{3}} \Pr\{E_{W^{1}}^{1}\}\) denotes \(\sum_{w_{3}=1}^{\varepsilon^{3}} \Pr\{E_{W^{1}}^{1}\}\), etc. It follows from Cover and Thomas [1, Th. 7.6.1, part 1] that

$$
\Pr\{E_{W^{1}}^{1}\} \leq \varepsilon
$$

$$
\Pr\{E_{W^{2}}^{2}\} \leq \varepsilon
$$

$$
\Pr\{E_{W^{3}}^{3}\} \leq \varepsilon
$$

for all $T$ large enough. Moreover, part 3 of this theorem implies that

$$
\Pr\{E_{w_{1}}^{1}\} \leq 2^{-T(1 + 3 - 1) - 3}\varepsilon + 2^{-T(1 - 3) - 3}\varepsilon
$$

$$
\Pr\{E_{w_{2}}^{2}\} \leq 2^{-T(1 + 2 - 1) - 3}\varepsilon
$$

$$
\Pr\{E_{w_{3}}^{3}\} \leq 2^{-T(1 + 2 - 1) - 3}\varepsilon
$$

for $w_{1} \neq W_{(1)}, w_{2} \neq W_{(2)},$ and $w_{3} \neq W_{(3)}$. If we now use a rate-matching code with

$$
R_{1} = C_{1} - 4\varepsilon,
R_{2} = C_{2} - 4\varepsilon,
R_{3} = C_{3} - 4\varepsilon
$$

then using (16) we obtain

$$
\Pr\{\hat{W}(\pi) \neq W\} \leq \varepsilon + 2^{T}R_{1}2^{-T(1 - 3) - 3}\varepsilon + \varepsilon + 2^{T}R_{2}2^{-T(1 - 3) - 3}\varepsilon
$$

$$
\leq 6\varepsilon
$$

for all $T$ large enough. There are $3! = 3! = 6$ possible permutations however. Therefore

$$
\Pr\{U_{\pi}^{1}, \hat{W}(\pi) \neq W\} \leq 36\varepsilon
$$

This implies that for all $T$ large enough there exist at least one code for which the error probability $\Pr\{\hat{W}(\pi) \neq W\}$ is smaller than $36\varepsilon$ for all 6 permutations $\pi$ when $\frac{1}{2}\varepsilon^{1}\log_{2}M = C_{1} + C_{2} + C_{3} - 12\varepsilon$. Consequently $C_{1} + C_{2} + C_{3}$ is achievable with a single code and a sequential decoding procedure.

**Comment:** Observe that this proof generalizes to arbitrary $S > 3$ but also to the case where $S = 2$. For $S = 2$ we can use a $(2^{2T(1+2)} R_{1}, 2^{2T(1+2)} R_{2})$-rate-matching code based on two mappings $f_{1}$ and $f_{2}$, with $R_{1} = C_{1} - 4\varepsilon$ and $R_{2} = C_{2} - 4\varepsilon$ which is very easy to construct as we shall see in the next section.

---

**VI. CONSTRUCTION OF RATE-MATCHING CODES**

In Section IV we have introduced rate-matching codes. We assumed there that rate-matching codes owning the properties stated in Definition 1 exist. Here we will show how such rate-matching codes can be constructed. We start with codes for $S = 3$.

**A. Construction Based on a Venn-Diagram for $S = 3$**

Although so far we have considered indices $i$ assuming values in $\{1, \ldots, M\}$ for some positive integer $M$, from now on we assume that the set of possible indices is $\{0, 1, \ldots, 2^{L} - 1\}$ where $L$ is a nonnegative integer. This implies that index $i$ is equivalent to some binary sequence $(b_{1}, b_{2}, \ldots, b_{L})$ in the sense that $i = \sum_{l=1}^{L} b_{l}2^{L-1}$. Observe that digits $B_{1}, B_{2}, \ldots, B_{L}$ in the binary sequence should be independent and uniformly distributed if the index $I$ must be uniform, and vice versa. Next we define the combined index $(i', i'')$ as the index that is equivalent to the concatenation $(b_{1}', b_{2}', \ldots, b_{L}', b_{1}'', b_{2}'', \ldots, b_{L}'')$ of the binary sequences $(b_{1}', b_{2}', \ldots, b_{L}')$ and $(b_{1}'', b_{2}'', \ldots, b_{L}'')$ that correspond to $i'$ and $i''$ respectively. Note that the combined index $(i', i'')$ assumes values in $\{0, 1, \ldots, 2^{L} - 1\}$. Three or more indices can be combined in a similar way.

In order to construct a rate-matching code, we consider 7 indices $i_{1}, i_{2}, i_{3}, i_{12}, i_{23}, i_{13},$ and $i_{123}$. Assume that indices $i_{1}, i_{2},$ and $i_{3}$ each correspond to sequences consisting of $L_{1}$ binary digits. Indices $i_{12}, i_{23},$ and $i_{13}$ correspond to sequences of $L_{2}$ binary digits, and index $i_{123}$ to a sequence of $L_{3}$ digits. The three message indices $w_{1}, w_{2},$ and $w_{3}$ are now obtained by combining the indices $i_{1}, i_{2}, \ldots, i_{123}$ in a "Venn-diagram"-manner, see Fig. 4, i.e.

$$
w_{1} = (i_{1}, i_{12}, i_{31}, i_{123})
$$

$$
w_{2} = (i_{2}, i_{23}, i_{12}, i_{123})
$$

$$
w_{3} = (i_{3}, i_{31}, i_{13}, i_{123})
$$

The construction results in a $(2^{2T_{1}}, 2^{2T_{2}} R_{1}, 2^{2T_{2}} R_{2}, 2^{2T_{3}} R_{3})$-rate-matching code with $R = R_{1} + R_{2} + R_{3}$ and

$$
TR_{1} = L_{1} + 2L_{2} + L_{3}
$$

$$
TR_{2} = L_{1} + L_{2}
$$

$$
TR_{3} = L_{1}
$$

for all nonnegative integers $L_{1}, L_{2},$ and $L_{3}$. The nonnegativity of the $L_{1}, L_{2},$ and $L_{3}$ now implies that the inequalities

$$
R_{1} \geq R_{2} \geq R_{3} \geq 0
$$

$$
R_{1} - R_{2} \geq R_{2} - R_{3}
$$

(34)
must hold. The last inequality in (34) now causes a problem if we want to achieve $C_1 + C_2 + C_3$ for capacities for which $C_1 - C_2 < C_2 - C_3$. A construction that is effective also for such capacities will be presented next however.

The Venn-diagram construction was proposed and investigated by the second author in [3]. Note that for $S = 2$ (i.e., the case with two parallel channels) the Venn-diagram method does not impose a constraint on $R_1$ and $R_2$ other than $R_1 \geq R_2$. Therefore the Venn-diagram construction is optimal for $S = 2$.

**B. Construction Based on Binary Codes for $S = 3$**

Consider a matrix of binary digits $\{b_{s,k}: s = 1, \ldots, 3, k = 1, \ldots, k_3\}$. Furthermore let $k_1 \geq k_2 \geq k_3 \geq 0$ for integers $k_1$, $k_2$, and $k_3$, see Fig. 5. The binary digits in the matrix relate to each other in the following way:

- The digits $b_{s,k}$, for $s = 1$ and $k = 1, \ldots, k_1$, for $s = 2$ and $k = 1, \ldots, k_2$, and for $s = 3$ and $k = 1, \ldots, k_3$ are information symbols. They can be chosen independently and together these digits represent the message index $m$.
- Note that in columns $k = 1, \ldots, k_3$ the digits $b_{1,k}$, $b_{2,k}$, and $b_{3,k}$ are information symbols.
- In columns $k = k_3 + 1, \ldots, k_2$ the digits $b_{1,k}$ and $b_{2,k}$ are information symbols. The digits $b_{3,k}$ are parity symbols, i.e.

$$b_{3,k} = b_{1,k} \oplus b_{2,k}$$

(35)

where $\oplus$ is modulo-2 addition.

- In columns $k = k_2 + 1, \ldots, k_3$ the digits $b_{1,k}$ are information symbols. The digits $b_{2,k}$ and $b_{3,k}$ are parities again, but now

$$b_{2,k} = b_{1,k}$$

$$b_{3,k} = b_{1,k}$$

(36)

Note that in columns $k = k_3 + 1, k_1$ we are using a length-three binary repetition code and each of the binary digits in such a column determines the other two digits in that column. In columns $k = k_3 + 1, k_2$ we apply a binary single-parity-check code and each pair of binary digits in such a column determines the remaining digit in that column. Columns $k = 1, k_3$ are uncoded, all eight digit-combinations are possible in such a column.

If we now define the indices as follows:

$$w_1 \triangleq \sum_{k=1}^{k_1} b_{1,k} 2^{k-1}$$

$$w_2 \triangleq \sum_{k=1}^{k_2} b_{2,k} 2^{k-1}$$

$$w_3 \triangleq \sum_{k=1}^{k_3} b_{3,k} 2^{k-1}$$

(37)

we have constructed a $(2^{TR_1}, 2^{TR_2}, 2^{TR_3})$-rate-matching code with $R = R_1 + R_2 + R_3$ and

$$TR_1 = k_1$$

$$TR_2 = k_2$$

$$TR_3 = k_3$$

(38)

for integer $k_1$, $k_2$, and $k_3$. Although $k_1 \geq k_2 \geq k_3 \geq 0$ implies that condition

$$R_1 \geq R_2 \geq R_3 \geq 0$$

(39)

should hold, this causes no problem. The consequence of this is that $C_1 + C_2 + C_3$ is also achievable with a single channel code together with a rate-matching code based on simple binary codes and sequential decoding.

**C. Construction of a Rate-Matching Code for $S > 3$**

If we consider the case where we have $S \geq 3$ parallel channels, then, to create a rate-matching code, for each $k = 1, \ldots, S$ we need codes of length $S$ with the property that any $k$ symbols of a codeword fully determine the remaining symbols. Codes that have this property are called maximum distance separable (MDS), see MacWilliams and Sloane [4, Ch. 11]. The symbols in these codes are not always binary, as in the previous subsection. Our construction is based on Reed–Solomon codes over $GF(2^m)$, where we take $m$ such that $2^m - 1 \geq S$. There exist $[2^m - 1, k, 2^m - k]$ Reed-Solomon codes for all $k = 1, \ldots, 2^m - 1$. Here $[n, k, d]$ refers to a linear code with length $n$, dimension $k$, and minimum Hamming distance $d$. These Reed-Solomon codes all have the MDS property and therefore any $k$ symbols may be taken as information symbols.

Take $k = 1, \ldots, S$ in what follows. It is a consequence of [4, Corollary 3, Ch. 11] that since MDS-code $[2^m - 1 + k, 2^m - k]$ exists over $GF(2^m)$ there exists another MDS-code $[2^m - 1 + j, k, 2^m - k - j]$ as long as $j \leq 2^m - 1 - k$. Just delete $j$ columns from the original generator matrix, then there are at least $k$ columns left. If we take $j = 2^m - 1 - S$ then we get an $[S, k, S - k + 1]$ MDS code over $GF(2^m)$. Since $S \geq k$ this is allowed.

Using a similar construction as in the previous subsection, based on the MDS property of the constituent codes, we obtain a $(2^{TR_1}, 2^{TR_2}, 2^{TR_3}, \ldots, 2^{TR_S})$-rate-matching code with $R = R_1 + R_2 + \cdots + R_S$ and

$$TR_1 = mk_1$$

$$TR_2 = mk_2$$

$$\cdots$$

$$TR_S = mk_S$$

(40)

for negative integers $k_1, k_2, \ldots, k_S$ satisfying $k_1 \geq k_2 \geq \cdots \geq k_S$; see Fig. 6. Therefore, again the only restriction for the rates is that

$$R_1 \geq R_2 \geq \cdots \geq R_S \geq 0$$

(41)

Therefore $\sum_{i=1}^{S} C_i$ is also achievable for $S > 3$ with a single channel code combined with an MDS-code based rate-matching code and sequential decoding.

**D. A Pseudo Rate-Matching Code Based on Joint Typicality**

In Section IV we have defined what we mean by a rate-matching code. Moreover in this section we have described how rate-matching codes can be constructed. Here we will consider special sets of jointly-typical sequences that have properties which are not as strict as those of rate-matching codes, but nevertheless these special sets can be used as

---

Fig. 5. A rate-matching code for three parallel channels.
(pseudo) rate-matching codes in our achievability proof. Consider three discrete random variables $U_1, U_2, U_3$ that have entropies $H(U_1) = H(U_2) = H(U_3) = \delta$, $H(U_1, U_2) = H(U_2, U_3) = H(U_3, U_1) = h_1 + h_2 + h_3$. Fix an integer sequence length $n$, a $\delta > 0$, and observe now that for all $i, j, k \in \{1, 2, 3\}$ and $i \neq j, j \neq k, k \neq i$

$$\left| A^\delta_{U_i} (U_{i1}, U_{ij}, U_{ik}) \right| \leq 2^{n(h_1 + \delta)}$$

$$\left| A^\delta_{U_j} (U_{j1}, U_{ji}, U_{jk}) \right| \leq 2^{n(h_2 + 2\delta)}$$

$$\left| A^\delta_{U_k} (U_{k1}, U_{ki}, U_{kj}) \right| \leq 2^{n(h_1 + h_2 + 2\delta)}$$

(42)

where $A^\delta_{U_i}$, $A^\delta_{U_j}$, and $A^\delta_{U_k}$ denote sets of (conditionally) typical sequences $u_i$, of length $n$, see Cover and Thomas [1, Sec. 15.2]. Note that from comparing (16) and (42) we may conclude that the sets containing sequences $u_1, u_2$, and $u_3$ satisfy inequalities that are similar to the equalities for the sets containing indices $u_1, u_2$, and $u_3$. Since the inequalities in (42) are upper bounds for the set cardinalities we could use the sequences $u_1, u_2$, and $u_3$ instead of $w_1, w_2$, and $w_3$ as indices to $2^{n(h_1 + \delta)}$ randomly chosen codewords in our achievability proof. If moreover we use sequence-triple $(u_1, u_2, u_3)$ as a replacement for message-index $w$ it follows immediately that our pseudo rate-matching code is one-to-one. A crucial observation is that

$$\left| A^\delta_{U_i} (U_1, U_2, U_3) \right| \geq 2^{n(h_1 + h_2 + h_3 - 2\delta)}$$

(43)

for all $n$ large enough. Therefore we can use this pseudo rate-matching code in our achievability proof to convey at least $TR = n[h_1 + h_2 + h_3 - 2\delta]$ bits to the receiver if we set $n[h_1 + \delta] = TR_1, n[h_2 + 2\delta] = TR_2$, and $n[h_3 + 2\delta] = TR_3$. A problem that still remains to be solved is to find probability distributions $P(u_1, u_2, u_3), u_1 \in \mathcal{U}_1, u_2 \in \mathcal{U}_2, u_3 \in \mathcal{U}_3$ with desired (conditional) entropies $h_1, h_2$, and $h_3$.

As a final remark we mention that the above method can also be used in a more direct way to transmit the output sequences $u_1, u_2$, and $u_3$ generated by a correlated i.i.d. source with generic random variables $U_1, U_2$, and $U_3$ as described before, to the receiver, over arbitrarily permuted channels if $T(C_1 - 4\epsilon) > n[h_1 + \delta], T(C_2 - 4\epsilon) > n[h_2 + 2\delta]$, and $T(C_3 - 4\epsilon) > n[h_3 + 2\delta]$. This technique is a kind of Slepian-Wolf [5] coding, since the encoders for the (dependent) separate sequences can operate independently of each other and yet reliable transmission is possible for total source entropy $h_1 + h_2 + h_3$ not larger than but arbitrarily close to the total channel capacity $C_1 + C_2 + C_3$, forgetting about the factor $T/n$ for a moment.

VII. AN APPLICATION

We will study an application in this subsection based on AWGN channels. Note that so far we have only considered discrete channels here. It is not hard to see that our results carry over to the AWGN case however.

Consider a communication system with two parallel AWGN channels that are described by the equations

$$Y_1 = A_1X_1 + N_1$$

$$Y_2 = A_2X_2 + N_2$$

(44)

where $X_1$ and $X_2$ are complex channel input variables, $N_1$ and $N_2$ are circularly symmetric complex Gaussian noise variables, and $A_1$ and $A_2$ are complex channel coefficients. We assume that $A_1$ and $A_2$ are circularly symmetric complex randomly chosen Gaussians with variance 1. The moduli of the channel coefficients therefore have a Rayleigh density, i.e.,

$$p_{|A_1|}(\alpha) = p_{|A_2|}(\alpha) = 2\alpha \exp(-\alpha^2), \text{ for } \alpha \geq 0.$$  

(45)

Moreover we suppose that the coefficients are generated independently of each other and of the channel inputs, and that they are constant over the duration of the code-sequences. The channel inputs are power-constrained, i.e., $E[|X_1|^2] \leq P$ and $E[|X_2|^2] \leq P$. The noise variables have variances $E[|N_1|^2] = E[|N_2|^2] = 1$.

Capacity is now achieved if both $X_1$ and $X_2$ are circularly symmetric complex Gaussians with total variance $P$. The capacity of a channel however depends on the channel coefficient, hence

$$C_1 = \log_2(1 + |A_1|^2 P)$$

$$C_2 = \log_2(1 + |A_2|^2 P).$$

(46)

Observe that therefore the capacities $C_1$ and $C_2$ are random variables. For $P = 48$ we have computed the probability density function of the pair $(C_1, C_2)$. A contour plot of this density can be found in Fig. 7. Note that the density is largest when both capacities are between 5 and 6 bits.

A) Now suppose first that we communicate independently over both channels and apply a code with rate $R = 2$ bits for each channel. Then communication can only be reliable if both capacities $C_1 \geq 2$ bits and $C_2 \geq 2$ bits. If this is not the case we speak about outage. The outage probability is defined as

$$P_{\text{out}}^A \triangleq 1 - \Pr \{(C_1 \geq 2) \land (C_2 \geq 2)\}$$

(47)
Note that the channels have a capacity larger than 2 bits if $|A_1|^2 \geq 1/16$ and $|A_2|^2 \geq 1/16$. Therefore the outage probability
\[ P_{\text{out}}^A = 1 - (\exp(-1/16))^2 = 0.1175. \]  

B) Next note that a total rate of 4 bits can also be achieved if we can transmit with rate $R_1 = 3.3$ bits over the strong channel and with rate $R_2 = 0.7$ bits over the weak channel. Note that since the transmitter does not know which channel is strong and which is weak we could use the signaling method that we have developed here. We may assume that the receiver (e.g., by applying pilots) knows the state of the channels however. Assuming that channel $i$ is the strong channel and channel $j$ is the weak channel, both inequalities $|A_i|^2 \geq 0.1844$ and $|A_j|^2 \geq 0.0130$ should be satisfied. Now for the outage probability we can write
\[
P_{\text{out}}^H = 1 - \Pr\{((C_1 \geq 0.7) \land (C_2 \geq 3.3)) \lor ((C_1 \geq 3.3) \land (C_2 \geq 0.7))\} = 1 - (e^{-0.0130})^2 + (e^{-0.0130} - e^{-0.1844})^2 = 0.0498 \]  
(49)

see again Fig. 7.

We may conclude that using our coding method for transmission over arbitrarily permuted parallel channels results in a smaller outage probability. The reason for getting an improvement is simply that we are more flexible. Condition $(C_1 \geq 2) \land (C_2 \geq 2)$ is just a special case of condition $(C_1 \geq 2-\delta) \land (C_2 \geq 2+\delta) \lor (C_1 \geq 2+\delta) \land (C_2 \geq 2-\delta))$.

Better results can be obtained for $\delta = 1.3$.

VIII. CHANNELS WITH DIFFERENT CAPACITY-AchieVING DISTRIBUTIONS

So far we have only focussed on parallel channels that all achieve capacity for the same input distribution. In the present section we will consider the case where these channels have different capacity-achieving distributions however. The result that we obtain here is stated in the next theorem.

**Theorem 2:** The capacity for transmission over $S$ arbitrarily permuted parallel channels equals
\[
C_H = \max_{Q(x)} \sum_{i=1}^{S} I(X; Y_i) \tag{50}
\]
where $I(X; Y_i)$ is as defined in (7). Note that all channels have an identical input alphabet $\mathcal{X}$.

Note that now $C_H$ is in general smaller than the sum of the capacities of each of the parallel channels. When there is a single distribution that achieves capacity for all parallel channels we get equality however.

The proof of this theorem actually consists only of a converse part. Achievability, both for the basic and the sequential case, follows immediately from the achievability proofs in Sections III and V if we replace $\{Q^*(x), x \in \mathcal{X}\}$ by the (or a) distribution that achieves the maximum in (50).

To prove the converse we first fix an $\varepsilon > 0$ and a block-length $T$. Now consider a permutation $\pi$ and a code that achieves $\Pr\{\hat{W}(\pi) \neq W\} \leq \varepsilon$. Then
\[
\log_2(\lambda(M)) \leq H(\hat{W}) = I(W; \hat{Y}_1, \hat{Y}_2, \ldots, \hat{Y}_S) + H(W|\hat{Y}_1, \hat{Y}_2, \ldots, \hat{Y}_S) \\
\leq I(X; \hat{Y}_1, \hat{Y}_2, \ldots, \hat{Y}_S) + H(W|\hat{W}(\pi)) \\
\leq \sum_{i=1}^{S} I(X_{\pi(i)}; Y_i) + H(W|\hat{W}(\pi)) \\
\leq \sum_{i=1}^{S} \sum_{t=1}^{T} I(X_{\pi(i),t}; Y_{i,t}) + 1 + \varepsilon \log_2(M). \tag{51}
\]
Note that in the last step we used Fano’s inequality. Rewriting (51) leads, for all permutations $\pi$, to
\[
\log_2(\lambda(M)) \leq \frac{1}{1-\varepsilon} \left[ \sum_{i=1}^{S} \sum_{t=1}^{T} I(X_{\pi(i),t}; Y_{i,t}) + 1 \right]. \tag{52}
\]
Since this code has to be good for all permutations, we can combine all these inequalities. We then obtain
\[
\frac{\log_2(\lambda(M))}{T} \leq \frac{1}{S^!} \sum_{i=1}^{S!} \frac{1}{1-\varepsilon} \left[ \sum_{i=1}^{S} \sum_{t=1}^{T} I(X_{\pi(i),t}; Y_{i,t}) + 1 \right] \\
= \frac{1}{1-\varepsilon} \sum_{i=1}^{S^!} \frac{1}{S^!} \left[ \sum_{i=1}^{S} \sum_{t=1}^{T} I(X_{\pi(i),t}; Y_{i,t}) + 1 \right] \\
\leq \frac{1}{1-\varepsilon} \left[ \sum_{i=1}^{S^!} \sum_{t=1}^{T} I(X_{i,t}; Y_{i,t}) + 1 \right]. \tag{53}
\]
The second inequality follows from the convexity of mutual information over the channel’s input distribution. We assume that for all $t = 1, \ldots, T$ the variables $X_{i,t}, X_{2i,t}, \ldots, X_{Si,t}$ are obtained from permuting $X_{1t}, X_{2t}, \ldots, X_{St}$ in all $S!$ ways, uniformly. This results in
\[
\Pr\{X_{i,t} = x_1, \ldots, X_{Si,t} = x_S\} = \frac{1}{S^!} \sum_{i=1}^{S!} \Pr\{X_{i,t} = x_1, \ldots, X_{Si,t} = x_S\} \tag{54}
\]
for all $x_1, x_2, \ldots, x_S \in \mathcal{X}$. This implies that
\[
\Pr\{X_{i,t} = x\} = \Pr\{X_{2i,t} = x\} = \cdots = \Pr\{X_{Si,t} = x\} = \frac{1}{S} \sum_{i=1}^{S} \Pr\{X_{i,t} = x\} \tag{55}
\]
for all $x \in \mathcal{X}$. Therefore $X_{1t}, X_{2t}, \ldots, X_{St}$ are all random variables with the same probability distribution and we denote them all by $Y_t$.

Also the third inequality follows from the convexity of mutual information over the channel’s input distribution. If we take
\[
\Pr\{X = x\} = \frac{1}{T} \sum_{i=1}^{T} \Pr\{X_i = x\} \tag{56}
\]
for all $x \in \mathcal{X}$, we get the third inequality.

The converse now ends in the standard way, i.e., by letting $T \to \infty$ and $\varepsilon \downarrow 0$.

IX. FINAL REMARKS

Our rate-matching code followed by the randomly chosen code resembles a Blokh-Zyablov generalized concatenated code [6]. In the Blokh-Zyablov construction outer codes are used with different dimensions but all having the same length, and a single inner code, see [7]. The column codes in our rate-matching construction in Sections VI-B and VI-C are equivalent to the outer codes. Our randomly generated code is the equivalent of the inner code in the Blokh-Zyablov construction. What makes our code construction special is the fact that our code is used to transmit efficiently over a number of arbitrary permuted parallel channels while the Blokh-Zyablov motivation was to find codes with good distance properties.

The methods that were proposed here have been used to improve the performance of V-BLAST systems proposed by Foschini [8], see
ACKNOWLEDGMENT

The authors wish to thank their colleagues Ludo Tolhuizen and Andries Hekstra, but especially the two anonymous reviewers, for valuable comments and insightful remarks. The remarks of one of the reviewers motivated us to investigate the case where the capacity-achieving input distributions for the parallel channels differ from each other.

REFERENCES