On the Capacity of Permanent Memory

CHRIS HEEGARD, MEMBER, IEEE

Abstract—Many forms of digital memory have been developed for the permanent storage of information. These include keypunch cards, paper tapes, PROMs, photographic film and, more recently, digital optical disks. All these “write-once” memories have the property that once a “one” is written in a particular cell, this cell becomes irreversibly set at one. Thus, the ability to rewrite information in the memory is hampered by the existence of previously written ones.

The problem of storing temporary data in permanent memory is examined here. Consider storing a sequence of $t$ messages $W_1, W_2, \ldots, W_t$ in such a device. Let each message $W_i$ consist of $k_i$ bits and let the memory contain $n$ cells. We say that a rate $t$-tuple $(R_1 = k_1/n, R_2 = k_2/n, \ldots, R_t = k_t/n)$ is achievable if we can store a sequence of messages at these rates for some $n$. The capacity $C^* t$ is the closure of the set of achievable rates. The capacity $C^*$ for an optical disk-type memory is determined. This result is related to the work of Rivest and Shamir.

A more general model for permanent memory is introduced. This model allows for the possibility of random disturbances (noise), larger input and output alphabets, more possible cell states, and a more flexible set of state transitions. An inner bound on the capacity region $C^*$ for this model is presented. It is shown that this bound describes $C^*$ in several instances.

I. INTRODUCTION

W E ARE interested in the temporary storage capacity of memories that have been developed to store permanent information. This class of “write-once” memories (WOMs) includes keypunch cards, paper tape, PROMs, photographic film and, more recently, digital optical (or video) disks. A keypunch card is a generic example of this type of memory. Binary data is represented on a card by associating the numbers zero and one with a blank or mark, respectively. Once a one (mark) is written on the card, that location becomes permanently associated with a one. Thus the ability to future write data on the card is hampered by the existence of previously written ones.

In a recent paper [1], Rivest and Shamir consider the possibility of rewriting information in permanent memory. They note the potential cost/performance of the new class of digital optical disks. The disks cost on the order of $100.00, have on the order of $10^{11}$ memory cells (the equivalent of 40 reels of magnetic tape), and have a high access rate. Rivest and Shamir ask the following question: If we would like to store a sequence of $t$ messages $W_1, W_2, \ldots, W_t$, each of which consists of $k$ bits, how many binary WOM cells $n*(k, t)$ would be required? Obviously $n*(k, t) \geq k$, but the interesting fact is that $n*(k, t)$ can be much less than the product $kt$. The motivating example is a code that can store two bits ($k = 2$), in three binary WOM cells ($n = 3$), twice ($t = 2$). Rivest and Shamir determine $n*(k, t)$ for modest values of $k$ and $t$, determine

$$C^*(t) = \lim_{k \to \infty} k/n*(k, t)$$

for small $t$ ($C^*(2) = 0.7728$), and show

$$C^*(t) \approx \frac{1}{t} \log_2 (1 + t)$$

for large $t$. The authors also show

$$\lim_{t \to \infty} t/n*(k, t) = 1$$

for every $k$ and conjectured $n*(k, t) = \max (i, kt/\log_2 (1 + i))$ for large $k$ and $t$.

In this paper we expand on some of the notions introduced in [1]. In the process, we answer many of the open questions suggested in the conclusion of [1]. We put these problems into a coding and information theory framework. In many ways, these questions are related to the problem of storing messages in defective computer memory [2]–[4]. References [2]–[4] concern the capacity of defective memory (typically, “stuck-at” cells) when information concerning the locations of the defects is available to the writer (encoder). The difference in our problem lies in the fact that the “defects” are introduced by the storage of previous messages in memory. When we store message $W_t$, we first read the memory to obtain this defect information. This “side” information is then used to encode the data in such a manner that the creation of new defects is minimized. In this way we better utilize the potential storage capacity of the memory.

We begin our discussion by looking at the Rivest–Shamir problem in a more general setting. Suppose that we would like to store a sequence of $t$ messages $W_1, W_2, \ldots, W_t$, where $W_i$ is a message consisting of $k_i$ bits. We say that a rate $t$-tuple $(R_1 = k_1/n, R_2 = k_2/n, \ldots, R_t = k_t/n)$ is achievable if we can store a sequence of messages at these rates for some $n$. The capacity $C^* t$ is the closure of the set of achievable rates. We determine $C^*$ for the Rivest–Shamir model and relate this to $C^*(t)$. We also establish that the maximum average rate achievable

$$C^*(t) = \max \left\{ \frac{1}{t} \sum_{j=1}^{t} R_j \right\}$$

is equal to $(1/t) \log_2 (1 + t)$ for all $t$.

We then develop a more general WOM model. This model allows for the possibility of random disturbances
HEEGARD: CAPACITY OF PERMANENT MEMORY

(noise), larger input and output alphabets, more possible cell states, and a more flexible set of state transitions. An inner bound on the capacity $C_\ast$ is established for this general WOM model. Although it is unlikely that the inner bound is the true capacity, the bound is tight in several special cases. It is shown that when the output and the next state coincide, and when the output is a deterministic function of the current state and input, the achievable rate region is optimum. It is also demonstrated that for the Rivest-Shamir model with binary symmetric noise at the input, the bound describes the capacity region $C_\ast$.

In a recent paper by Wolf, Wyner, Ziv, and Korner [5], other generalizations of the Rivest-Shamir problem are studied. In a framework derived from Heegard and El Gamal [4], and Wolf et al. consider the $\varepsilon$-error capacity of deterministic, binary WOMs. As in [4], they study the WOM problem by considering the present state of the memory as side information available to the writer (encoder) and/or reader (decoder). Four cases are considered: 1) both encoder and decoder informed, 2) only encoder informed, 3) only decoder informed and 4) both encoder and decoder uninformed. Case (2) corresponds to our Theorem 4 when $\alpha = \beta = 0$. The most difficult case derived in [5] seems to be the last, where it is shown that

$$1 - R_2 \leq c_\ast(t) \leq \frac{1}{\ln 2} \log \left( \frac{\log 2}{\ln 2} \right)^{-1/2} \cdot \frac{1}{\ln 2} \log \left( \frac{\log 2}{\ln 2} \right)^{-1/2}$$

is the set of vectors compatible with $y$, i.e., $C(y)$, are those vectors that can be written on a disk that currently reads $y$. Note that such a choice of $x_3$ can be written on the disk undistorted and can be correctly decoded by recognizing its membership in $A^{x_3}$. Of course, this requires that $A_3$ not be empty. Since [6, p. 310]

$$|B_i| = \sum_{i=1}^{n} \binom{n}{i} \geq \frac{1}{\sqrt{\pi n e}} \left( \frac{1}{e} \right)^{n} 2^{n h(\varepsilon)}$$

(whreh $h(x) = -x \log_2 x - (1 - x) \log_2 (1 - x)$ is the binary entropy function), we can argue that for sufficiently large $n$, there exists a partitioning of $C_1$ with $A_3 \neq \emptyset$ for every $w \in \{1, 2, \ldots, 2^n R_1\}$, provided $R_1 < h(\varepsilon)$.

To store message $W_1 \in \{1, 2, \ldots, 2^n R_1\}$, simply choose any vector $x_1 \in A_3$, and write it on the disk. The value of $W_1$ can easily be obtained by recognizing that the vector written on the disk belongs to subset $A_3$. Of course, this requires that $A_3$ not be empty. Since [6, p. 310]

$$|B_i| = \sum_{i=1}^{n} \binom{n}{i} \geq \frac{1}{\sqrt{\pi n e}} \left( \frac{1}{e} \right)^{n} 2^{n h(\varepsilon)}$$

is the set of vectors compatible with $y$, i.e., $C(y)$, are those vectors that can be written on a disk that currently reads $y$. Note that such a choice of $x_3$ can be written on the disk undistorted and can be correctly decoded by recognizing its membership in $A_3$. The encoding of $W_1$ is successful whenever $A_3 \cap C(x_3)$ is nontrivial. We shall now use a random partition argument to show that when $R_1 < 1 - \varepsilon$ and $n$ is sufficiently large, there exists a partitioning of $C_1$ for which

$$A_3 \cap C(y) \neq \emptyset \quad \text{for every } w \in \{1, 2, \ldots, 2^n R_1\} \quad \text{and } y \in B_i.$$

Randomly partition $C_1$ into $2^n R_1$ subsets of equal size,

$$|A_3| = |C_1| 2^{-n R_1}.$$
vanishes for increasing $n$ when $R_2 < 1 - \epsilon$. Furthermore,

$$P \left( \text{there exists } w \in \{1, 2, \ldots, 2^{nR_2} \} \right)$$

and $y \in B_t$ with $A_w \cap C(y) = \emptyset$

$$= P \left( \bigcup_{w=1}^{2^{nR_2}} \bigcup_{y \in B_t} \left( A_w \cap C(y) = \emptyset \right) \right)$$

$$\leq \sum_{w=1}^{2^{nR_2}} \sum_{y \in B_t} P \left( A_w \cap C(y) = \emptyset \right)$$

$$\leq 2^{n(R_2 + h(\epsilon))} e^{-(2^{nR_2} - 1)},$$

where we use the fact that $|B_t| \leq 2^{n^2(\epsilon)}$ [6, p. 310]. We see that for $R_2 < 1 - \epsilon$ and large $n$, this probability becomes negligible. This shows not only the existence of the desired partitions of $C_2$ but also the fact that for large $n$, almost every partition satisfies the requirement that $A_w \cap C(y) \neq \emptyset$ for every $w \in \{1, 2, \ldots, 2^{nR_2} \}$

Combining these results, we see that there exists a partition code that allows us to store $W_1$ followed by $W_2$ for any rate pair $(R_1, R_2)$ in the set

$$\left\{ (R_1, R_2) \in \mathbb{R}^2 \mid R_2 < 1 - \epsilon, 0 \leq \epsilon \leq 1/2 \right\}$$

(2)

Let $C^*_2$ denote the closure of this set (2). Note that the set (1) defined by dividing the disk is strictly contained in $C^*_2$ (see Fig. 1). We will presently show that it is not possible to do better than (2); thus we refer to $C^*_2$ as the zero-error capacity region for a sequence of two writes. Similar reasoning can be used to determine the zero-error capacity region $C^*_t$ for any finite value for the sequence length $t$

Before we state this result as a theorem, consider the following definitions.

The binary "or" operator $\lor$ is defined by

$$x \lor y = \begin{cases} 0, & \text{if } x = y = 0 \\ 1, & \text{otherwise} \end{cases}$$

A deterministic binary WOM (e.g., a nonerasable disk) can be modeled by $y = x \lor s$, where $y, x, s \in \{0, 1\}^n$, and the or operator is performed component-wise. We can interpret the $x$ vector as the WOM input and the $s$ vector as the present WOM state (the vector already written on the disk). The $y$ vector is both the WOM output and the next state of the disk.

A binary $(n, R_1, R_2, \ldots, R_t)$ code consists of $t$ encoding functions

$$f_i : \{1, 2, \ldots, 2^{nR_i} \} \times \{0, 1\}^n \to \{0, 1\}^n \quad 1 \leq i \leq t,$$

and $t$ decoding functions

$$g_i : \{0, 1\}^n \to \{1, 2, \ldots, 2^{nR_i} \} \quad 1 \leq i \leq t.$$  

(Note that $f_i$ need only be defined for those values of the second argument that lie in the range of $f_{i-1}$.) A code has no errors if for every sequence $w_1, w_2, \ldots, w_t,$

$$w_i = g_i (f_i (w_1, s_{i-1}) \lor s_{i-1}) \quad \text{for } i = 1, 2, \ldots, t$$

where $s_0 = 0$, $s_i = f_i (w_i, s_{i-1}) \lor s_{i-1}$.

We now argue that we cannot do better than $C^*_t$ when $t = 2$. Suppose that we have an encoder $(f_1(w_1), f_2(w_2, f_1(w_1)))$ that maps a message pair $(w_1, w_2)$ onto a pair of binary $n$-vectors and a decoder $(g_1, g_2)$ that maps these binary vectors onto an estimate of the message pair. We can assume without loss of generality that $f_2(w_2, f_1(w_1))$ is compatible with $f_1(w_1)$ (i.e., $f_2(w_2, f_1(w_1)) \in C(f_1(w_1))$). Let

$$nc = \max_{w \in \{1, 2, \ldots, 2^{nR_1} \}} ||f_1(w)||$$

be the maximum Hamming weight of a vector produced by the encoder $f_1$. Since $f_1$ must be one-to-one,

$$2^{nR_1} \leq \sum_{i=0}^{n} \binom{n}{i} \leq 2^{nh(\epsilon)}$$

or $R_1 \leq h(\epsilon)$. The second encoder, $f_2$, must also be one-to-one, in the first argument. Let $w$ be any message achieving $||f_1(w)|| = nc$. Then

$$2^{nR_2} \leq |C(f_1(w))| - 2^{n(1-\epsilon)}$$

or $R_2 \leq 1 - \epsilon$. Thus $(R_1, R_2) \in C^*_t$.

A rate $t$-tuple $(R_1, R_2, \ldots, R_t)$ is said to be zero-error achievable if and only if there exists an error-free $(n, R_1, R_2, \ldots, R_t)$ code for some $n$. The closure of the set of zero-error achievable rates is called the zero-error capacity region $C^*_t$.

**Theorem 1**: The zero-error capacity region for the deterministic binary WOM is the convex region

$$C^*_t = \left\{ (R_1, R_2, \ldots, R_t) \in \mathbb{R}^t \mid R_1 \leq h(\epsilon_1), \right.$$  

$$R_2 \leq (1 - \epsilon_1) h(\epsilon_2),$$

$$\vdots$$

$$R_{t-1} \leq (1 - \epsilon_1)(1 - \epsilon_2) \cdots (1 - \epsilon_{t-2}) h(\epsilon_{t-1}),$$

$$R_t \leq (1 - \epsilon_1)(1 - \epsilon_2) \cdots (1 - \epsilon_{t-1}),$$

where $0 \leq \epsilon_1, \epsilon_2, \ldots, \epsilon_{t-1} \leq 1/2$. \(\square\)

Up to this point, we have assumed that the generation number, i.e., the time index of the current message is known during both the encoding and decoding stages of storage, e.g., for $t = 2$ we know if $W_1$ or $W_2$ is stored. The
problem as originally formulated [1] did not have this assumption.

We define $C^*_0(t)$ as the zero-error, fixed-rate capacity when the generation number is not explicitly known by the encoder and decoder. The following argument will show

$$C^*_0(t) = \max_{(R_1, R_2, \ldots, R_t) \in C^*_1} \frac{1}{t} \sum_{j=1}^{t} R_j$$

is the maximum average capacity. We have the following corollary:

**Corollary:** For a deterministic binary WOM,

$$C^*_0(t) = \log_2 (1 + t).$$

Proving this result involves finding $\epsilon_1, \epsilon_2, \ldots, \epsilon_{t-1}$ that maximize

$$h(\epsilon_1) + (1 - \epsilon_1) h(\epsilon_2) + \cdots + (1 - \epsilon_1)(1 - \epsilon_2) \cdots (1 - \epsilon_{t-1}).$$

The solution is to set $\epsilon_{i-1} = 1/(2 + i)$. Thus, for example, in the $t = 2$ case, we find that $\epsilon = 1/3$ will maximize $R_1 + R_2$ in $C^*_0$ and that $C^*(2) = 1/2 \log(3)$. Note that $C^*_0(t) < C^*(t)$ for $t > 1$.

### III. ON THE CAPACITY OF A NOISY WOM

We have developed the notions of coding and capacity for the storage of a sequence of messages on a nonerasable disk. The operation of the individual cells of the disk could be described as follows. Initially, a binary letter applied to the input of the cell is faithfully reproduced at the output. However, once the value of the stored letter is a one, the operation of the cell is altered. From this point forward, the output of the cell becomes fixed at one. Thus we may describe the cell as a binary device with two states (the output equal the input state and the stuck-at-one state). Each cell begins in the first state and permanently transfers to the second state once a one is stored.

We now introduce a general memory cell model to allow for the possibility of random disturbances (noise), larger input and output alphabets, more cell states, and a more flexible set of state transitions. The addition of noise makes it likely that the zero-error capacity is not a useful notion (i.e., it is trivial). It is, in these cases, more meaningful to determine the $\epsilon$-error capacity of the memory (i.e., “At which rates can the probability of error be made arbitrarily small?”).

Consider the following definitions. An $(X, S, Y, p_0(s, y), p(s^+, y|x, s))$ generalized discrete memoryless WOM consists of three alphabets, $X$, $Y$ and $S$; a probability distribution $p_0(s, y)$ on the letters of $S \times Y$, and a conditional probability distribution $p(s^+, y|x, s)$ on the letters of $S \times Y$ conditioned on the letters of $X \times S$. We may interpret the cells as having an input space $X$, a state space $S$ and an output space $Y$.

For a memory consisting of $n$ cells, an initial state vector $\mathcal{S}_0 \subseteq S^n$ and output vector $\mathcal{Y}_0 \subseteq Y^n$ are generated according to the product distribution

$$P(\mathcal{S}_0 = s, \mathcal{Y}_0 = y) = \prod_{j=1}^{n} p_0(s_j, y_j).$$

Thus, initially each cell of the memory independently
chooses a state \( s \in S \) and output \( y \in Y \) with probability \( p_0(s, y) \). When an input vector \( x \in X^n \) is stored in a memory with state vector \( s^+ \in S^n \) and output vector \( \mathcal{V} \in Y^n \) are obtained according to the product distribution

\[
P(s^+, \mathcal{V} = y | x, s) = \prod_{j=1}^{n} p(s_j^+, y_j | x_j, s_j).
\]

Thus, for each cell with state \( s \in S \) and input \( x \in X \), the next state \( s^+ \in S^n \) and output \( y \in Y \) are independently selected with probability

\[
p(s^+, y | x, s) = \prod_{j=1}^{n} p(s_j^+, y_j | x_j, s_j).
\]

Thus, for each cell with state \( s \in S \) and input \( x \in X \), the next state \( s^+ \in S^n \) and output \( y \in Y \) are independently selected with probability

\[
P(s^+, y | x, s) = \prod_{j=1}^{n} p(s_j^+, y_j | x_j, s_j).
\]

For example, the deterministic binary WOM has alphabets \( X = Y = S = \{0, 1\} \), initial distribution \( p_0(0, 0) = 1 \), and a transition and output distribution

\[
P(W_0, 0) = P(W_1) = P(L_0, 1)
\]

\[
= p(1, 1 | 1, 1) = 1.
\]

In this case, the output is the next state and a function of the current state and input.

An \((n, R_1, R_2, \ldots, R_t, \delta)\) code consists of \( t \) encoding functions

\[
f_i: \{1, 2, \ldots, 2^{nR_i}\} \times Y^n \rightarrow X^n
\]

and \( t \) decoding functions

\[
g_i: Y^n \rightarrow \{1, 2, \ldots, 2^{nR_i}\}.
\]

Let \( W_1, W_2, \ldots, W_t \) be a sequence of independent messages with \( W_i \) uniformly distributed over the set \( U_i \in \{1, 2, \ldots, 2^{nR_i}\} \). A sequence of input, state, and output vectors \((S_0, Y_0), (S_1, Y_1), (S_2, Y_2), \ldots, (S_t, Y_t, \mathcal{V}_t)\) is obtained, where for \( 1 \leq i \leq t \), \( \mathcal{V}_i = f_i(W_i, \mathcal{V}_{i-1}) \). The \( i \)th probability of error is defined as

\[
P_i^e \equiv P(g_i(\mathcal{V}) \neq W_i).
\]

The (worst case) probability of error is

\[
\delta = \max_{1 \leq i \leq t} P_i^e.
\]

A rate \( t \)-tuple \((R_1, R_2, \ldots, R_t)\) is said to be \( \epsilon \)-achievable if for any \( \epsilon > 0 \) there exists an \((n, K_1, K_2, \ldots, K_t, \delta)\) code for some \( n \) with \( \delta < \epsilon \). The closure of the set of \( \epsilon \)-achievable rates \( C^*\) is called the \( \epsilon \)-error capacity region. \( C^*_{\epsilon}(t) \) is defined as the least upper bound on the set of \( \epsilon \)-achievable rates for a fixed encoder and decoder

\[
f: \{1, 2, \ldots, 2^{nR}\} \times Y^n \rightarrow X^n
\]

\[
g: Y^n \rightarrow \{1, 2, \ldots, 2^{nR}\}.
\]

(Not that these functions are not allowed to depend on the generation number of the message.) Thus \( C^*_{\epsilon}(t) \) is referred to as the fixed-rate capacity. Similarly, \( C^*(t) \) is defined as the maximum average rate that is \( \epsilon \)-achievable:

\[
C^*(t) = \max_{(R_1, R_2, \ldots, R_t) = C^*_{\epsilon}(t)} \frac{1}{t} \sum_{j=1}^{t} R_j.
\]

Let us consider how we might extend the idea of a partition code to a memory consisting of \( n \) \((X, S, Y, p_0(s, y), p(s^+, y | x, s))\) cells. Before we begin, we will need the notion of \( \epsilon \)-typical sets. Fix a small \( \epsilon > 0 \), and let \((\mathcal{X}, \mathcal{Y})\) be a pair of independent, identically distributed (i.i.d.) random vectors

\[
P(\mathcal{X} = x, \mathcal{Y} = y) = \prod_{j=1}^{n} p(x_j, y_j).
\]

Then the set of \( \epsilon \)-typical \( x \) vectors is defined as

\[
T_{\epsilon}(X) = \left\{ x \in X^n : \frac{1}{n} \sum_{j=1}^{n} 1_{x_j} - p(x) < \epsilon \right\}
\]

for every \( x \in X \),

where \( 1_{x_j} = 1 \) if \( x_j = x \) and 0 otherwise. This is the set of sequences for which the empirical frequency is within \( \epsilon \) of the probability \( p(x) \) for every letter \( x \in X \). We can similarly define the set of jointly \( \epsilon \)-typical vectors \( T_{\epsilon}(X,Y) \) and the set \( T_{\epsilon}(Y|x) \) of vectors \( y \in T_{\epsilon}(Y) \) that are jointly \( \epsilon \)-typical with a given vector \( x \in X^n \). (A complete discussion of \( \epsilon \)-typical vectors can be found in [7,8].) We shall need the following facts:

1) If \( \mathcal{X} \) is randomly chosen, then

\[
P(x \in T_{\epsilon}(X)) \rightarrow 1 \quad \text{as} \quad n \rightarrow \infty.
\]

2) If \( x \in T_{\epsilon}(X) \) and \( y \) is independently chosen according to the marginal distribution for \( \mathcal{X} \), then

\[
2^{-n(I(X,Y)+\lambda)} \leq P(\mathcal{Y} \in T_{\epsilon}(Y|x)) \leq 2^{-n(I(X,Y)-\lambda)}
\]

for some \( \lambda(e) > 0 \) with \( \lambda \rightarrow 0 \) as \( \epsilon \rightarrow 0 \) (note that \( I(X,Y) \) is the mutual information).

Let \( t = 2 \) and consider the following random partition argument. We will show that for certain values of \((R_1, R_2)\), we can randomly construct an encoder \((f_1, f_2)\) and decoder \((g_1, g_2)\) that will, on the average, have a probability of error that vanishes with increasing \( n \). This will prove the \( \epsilon \)-achievability of these rates.

Fix \( \epsilon > 0 \) and let \( U_i \in U \) be an auxiliary random variable. Choose a conditional distribution \( p_1(u|x,y) \) on \( U \times X \) conditioned on \( Y \). Let \( p_1(u) \) be the marginal distribution of \( U_i \) under the joint distribution \( p_0(s, y)p_1(u | x, y) \) of the random variables \((S_0, Y_0, U_i, X_i)\). Independently choose a set of \( 2^{nR_i} \) vectors according to the uniform distribution over the set \( T_{\epsilon}(U_i) \). Call this new set \( C_i \). Next, randomly partition \( C_i \) into \( 2^{nR_i} \) equal size subsets

\[
\{ A_{i1}, A_{i2}, \ldots, A_{i2^{nR_i}} \}.
\]

Let \( W_i \in \{1, 2, \ldots, 2^{nR_i}\} \) be the first message to be stored. An initial output \( \mathcal{V}_0 \) is read from the memory. Since \( \mathcal{V}_0 \) is correlated with the initial state of the memory \( S_0 \), the \( \mathcal{V}_0 \) vector can be useful in storing \( W_1 \). This is done by choosing a vector

\[
\mathcal{V}_1 \in A_{W_1} \cap T_{\epsilon}(U_i | \mathcal{V}_0).
\]

If such a vector exists, then randomly choose a vector from the set \( T_{\epsilon}(X|\mathcal{V}_1, \mathcal{V}_0) \) and write this vector into the memory. The encoding is successful whenever

\[
A_{W_i} \cap T_{\epsilon}(U_i | \mathcal{V}_0).
\]
HEEGARD: CAPACITY OF PERMANENT MEMORY

is not empty. Now
\[ P\left( A_{W_1} \cap T_e(U_1|\mathcal{Y}_0) = \emptyset \right) \]
\[ \leq P\left( \mathcal{Y}_0 \notin T_e(Y_0) \right) \]
\[ + P\left( A_{W_1} \cap T_e(U_1|\mathcal{Y}_0) = \emptyset | \mathcal{Y}_0 \in T_e(Y_0) \right). \]

Since \( P(\mathcal{Y}_0 \in T_e(Y_0)) \to 1 \) for \( \epsilon > 0 \) and large \( n \), we need to find a bound on the second term:

\[ P(A'_{W_1} \cap T_e(U_1|\mathcal{Y}_0) = \mathcal{Y}_0 \in T_e(U_1|\mathcal{Y}_0)) \]
\[ \leq \left( 1 - P\left( u \in T_e(U_1|\mathcal{Y}_0) | \mathcal{Y}_0 \in T_e(Y_0) \right) \right)^{A_{W_1}} \]
\[ \leq \left( 1 - 2^{-n(I(U_1; Y_0) + \lambda)} \right)^{2^{nR_1}} \]
\[ \leq \exp\left\{ -2^{-(Q_1 - R_1 - I(U_1; Y_0) - \lambda)} \right\}. \]

Thus if \( Q_1 - R_1 > I(U_1; Y_0) \), \( \epsilon \) is sufficiently small, and \( n \) is large, then the encoding will be successful with high probability.

To decode \( W_1 \), we first read \( \mathcal{Y}_1 \) from the memory. Then we look for a unique estimate \( \hat{a}_i \in C_{\mathcal{Y}_1} \cap T_e(U_1|\mathcal{Y}_1) \) of the vector \( a_i \). If such an estimate exists and it belongs to the \( A_{k} \) subset of \( C_{\mathcal{Y}_1} \), then we set \( I(\hat{a}_i = k) \).

\[ P(\hat{a}_1 \neq a_1) \leq P(\hat{a}_1 \neq a_1) \leq P(\mathcal{Y}_1 \notin T_e(U_1|\mathcal{Y}_1)) \]
\[ + P(\text{there exists a } u \neq \mathcal{Y}_1, \]
\[ u \in C_{\mathcal{Y}_1} \cap T_e(U_1|\mathcal{Y}_1), \]
\[ (\mathcal{Y}_1, a_1) \in T_e(U_1, Y_1)). \]

Again the first term will approach zero for large \( n \) and positive \( \epsilon \). The second term can be expressed as

\[ P\left( \bigcup_{u \in C_{\mathcal{Y}_1}} \left\{ u \in T_e(U_1|\mathcal{Y}_1) | (\mathcal{Y}_1, a_1) \in T_e(U_1, Y_1) \right\} \right) \]
\[ \leq \sum_{u \in C_{\mathcal{Y}_1} \prod_{u \neq \mathcal{Y}_1}} P(u \in T_e(U_1|\mathcal{Y}_1)(\mathcal{Y}_1, a_1) \in T_e(U_1, Y_1)) \]
\[ \leq 2^{n(Q_1 - R_1 - I(U_1; Y_0) - \lambda)}. \]

This probability will become negligible for large \( n \) when \( Q_1 < I(U_1; Y_1) \) and \( \epsilon \) is small. Combining this bound with the previous \( Q_1 - R_1 > I(U_1; Y_0) \), we conclude that rates \( R_1 < I(U_1; Y_1) - I(U_1; Y_0) \) are \( \epsilon \)-achievable.

A similar construction can be used to encode and decode \( W_2 \), since for large \( n \) with high probability \( (\mathcal{Y}_2, a_2, a_1) \in T_e(U_2, S_2, Y_2, X_2) \), we may be able to get a better estimate of the state \( a_2 \) with high probability. In this case, we choose a conditional distribution \( p_2(u^+, x|u, y) \) on \( U \times X \) conditioned on \( U \times Y \). The marginal \( p_2(u) \) of the random variable \( U_2 \) under the joint distribution

\[ p_0(s, y) p_1(u, y \mid y) p(s^+, y^+ \mid x, s) p_2(u^+, x^+ \mid y, y^+) \]

of the random variables \( (S_0, Y_0, U_1, X_1, S_1, Y_1, U_2, X_2, S_2, Y_2, \ldots, U_n, X_n, S_n, Y_n) \) is used to choose \( 2^{nQ_2} \) vectors \( C_2 \subset T_e(U_2) \). Then \( C_2 \) is partitioned into \( 2^nR_2 \) subsets of equal size

\[ \{ A_1^2, A_2^2, \ldots, A_{2^nR_2}^2 \}. \]

To encode \( W_2 \), choose \( \mathcal{Y}_2 \in A_{W_2} \cap T_e(U_2|\mathcal{Y}_2, \mathcal{Y}_1) \) and store a vector \( Z_2 \in T_e(U_2|\mathcal{Y}_2, \mathcal{Y}_1) \). This will be possible, with high probability, for \( Q_2 > R_2 > I(U_2; U_1, Y_1) \), small \( \epsilon \), and large \( n \). To decode \( W_2 \), find a unique \( \hat{a}_2 \in C_2 \cap T_e(U_2|\mathcal{Y}_2) \). If \( \hat{a}_2 \neq a_2 \), then \( W_2 = k \). The probability of a decoder error will be small for \( Q_2 < I(U_2; Y_2) \). Thus rates pairs \( (R_1, R_2) \) are \( \epsilon \)-achievable for

\[ R_1 < I(U_1; Y_1) - I(U_1; Y_0) \]
\[ R_2 < I(U_2; Y_2) - I(U_2; U_1, Y_1). \]

The following theorem extends this argument to any finite \( t \).

**Theorem 2:** An achievable rate region. Fix \( t, (X, S, Y, p_0(s, y), p(s^+, y|x, s)) \), and an auxiliary alphabet \( U \). Choose \( t \) conditional distributions

\[ p_1(u, x|y), p_2(u^+, x|u, y), \ldots, p_t(u^+, x|u, y) \]

for \( U \times X \) conditioned on \( U \times Y \). Let the joint distribution of the random variables \( (S_0, Y_0, U_1, X_1, S_1, Y_1, U_2, X_2, S_2, Y_2, \ldots, U_n, X_n, S_n, Y_n) \) take the form

\[ p_0(s_0, y_0) p_1(u_1, x_1|y_0) p(s_1, y_1|x_1, y_0) \]
\[ \cdot \prod_{i=2}^t p_i(u_i, x_i|u_{i-1}, y_{i-1}) p(s_i, y_i|x_i, s_{i-1}). \]

Then \( R_1, R_2, \ldots, R_t \in C_t^* \) if

\[ R_i < I(U_i; Y_i) - I(U_i; U_{i-1}, Y_{i-1}). \]

We note that the region described by Theorem 2 may not be convex; it is easy to show that convex combinations of rates described by Theorem 2 are also achievable.

Although it is unlikely that the region described by Theorem 2 is the capacity region \( C_t^* \), the following theorems demonstrate that for nontrivial cases this region is optimum. First we note that if we choose conditional distributions \( p_i(u^+, x|u, y) \) that do not depend on \( Z_i \) we get the following.

**Corollary:** Fix \( t, (X, S, Y, p_0(s, y), p(s^+, y|x, s)) \), and an auxiliary alphabet \( U \). Choose \( t \) conditional distributions

\[ p_1(u, x|y), p_2(u^+, x|u, y), \ldots, p_t(u^+, x|u, y) \]

for \( U \times X \) conditioned on \( U \times Y \). Let the joint distribution of the random variables \( (S_0, Y_0, U_1, X_1, S_1, Y_1, U_2, X_2, S_2, Y_2, \ldots, U_n, X_n, S_n, Y_n) \) take the form

\[ p_0(s_0, y_0) \prod_{i=1}^t p_i(u_i, x_i|y_{i-1}) p(s_i, y_i|x_i, s_{i-1}). \]

Then \( R_1, R_2, \ldots, R_t \in C_t^* \) if for \( i = 1, 2, \ldots, t \)

\[ R_i < I(U_i; Y_i) - I(U_i; U_{i-1}, Y_{i-1}). \]

We note that in general the (convex hull of the) rate region described by the corollary is a subset of the (convex hull of the) rate region of Theorem 2. However, when the output
and the state are the same (i.e., $S = Y$ and $S_i = Y$, $0 \leq i \leq t$), the rates achieved by the corollary are equivalent. This is intuitively reasonable. (To see that this is the case, use the chain rule for mutual information to write

$$I(U_i; Y) = I(U_i; U_{i-1}, Y_{i-1}) = I(U_i; U_{i-1}, Y) - I(U_i; U_{i-1}|Y).$$

Then note that $I(U_i; Y) - I(U_i; Y_{i-1})$ is only a function of

$$p(u_i, x_{i|Y_{i-1}}) - \sum_{u_{i-1} \in U} p(u_{i-1|Y_{i-1}}) p(u_i, x_{i|u_{i-1}, Y_{i-1}})$$

and not $p(u_i, x_{i|u_{i-1}, Y_{i-1}})$. Finally, note that

$I(U_i; U_{i-1}|Y_{i-1}) \geq 0$.)

If, in addition to $S = Y_i$, the output is a deterministic function of the input and state (i.e., $Y_i = e(X_i, Y_{i-1})$, then the corollary characterizes $C_i^*$.)

Theorem 3: The capacity of deterministic WOMs. Fix $i$ and let $(X_i, Y_i, p_i(y), e(x, y))$ describe a deterministic, memoryless WOM. Then

$$C_i^* = \{(R_1, R_2, \ldots, R_i) \in R^i_+ \mid R_i \leq H(Y_i|Y_{i-1}), 1 \leq i \leq t\},$$

where the joint distribution of the random variables $(Y_0, X_1, Y_1 = e(X_0, Y_0), X_2, Y_2 = e(X_2, Y_1), \ldots, X_i, Y_i = e(X_i, Y_{i-1}))$ is described by

$$p_0(y_0) p_1(x_1|y_0) p_2(x_2|y_1) \cdots p_i(x_i|y_{i-1}).$$

The region described by Theorem 3 is convex since the conditional entropy $H(Y_i|Y_{i-1})$ is a concave function of the joint distribution of $(Y_i, Y_{i-1})$. The achievability of the interior of this region follows by setting $U_i = Y_i$ in the corollary to Theorem 2. Next we will sketch the converse.

Fix a small $\varepsilon > 0$, and suppose there exists an $(n, R_1, R_2, \ldots, R_i, \delta)$ code with $\delta < \varepsilon$. Let $\Psi_i = e(f_i(W_i, \Psi_{i-1}), \Psi_{i-1})$, where $W_i, W_{i-1}, \ldots, W_1$ is a sequence of independent messages with $W_i$ uniformly distributed over the set $W_i \in \{1, 2, \ldots, 2^{nR_i}\}$. By Fano's inequality,

$$H(W_i|\Psi_i) \leq h(\varepsilon) + nR_i = n\theta_n,$$

where $\theta_n = (1/n)h(\varepsilon) + eR_i \to 0$ as $e \to 0$. Then

$$nR_i = H(W_i) = H(W_i|\Psi_{i-1}) \leq I(W_i, \Psi_{i-1}) + n\theta_n \leq \sum_{j=1}^{n} H(Y_i|Y_{i-1}^j) + \theta_n,$$

where $Y_i^j$ is the $j$th component of $\Psi_i$, etc. Finally, by the concavity of entropy, we can find random variables $(Y_i^*, Y_{i-1}^*)$ satisfying

$$R_i \leq H(Y_i^*|Y_{i-1}^*) \equiv \theta_n.$$
It is for this reason that the converse, which is given in the Appendix, holds.

For this model, we can also determine $C_0(t)$ and $C(t)$.

**Corollary:** For a binary WOM with binary symmetric input noise, $C_0(t)$ can be obtained recursively,

$$C_0(t) = (1 - \beta)(1 - h(\alpha)),$$

$$C_0(t+1) = (1 - \beta)\left\{h\left(\frac{z}{C_0(t)}\right) - z - h(\alpha)\right\}.$$ 

This result involves finding $\epsilon_1, \epsilon_2, \cdots, \epsilon_{t-1}$, which are the solution to

$$h(\alpha * \epsilon_1) - h(\alpha) = (1 - \alpha * \epsilon_1)(h(\alpha * \epsilon_2) - h(\alpha)) = \cdots = (1 - \alpha * \epsilon_1)(1 - \alpha * \epsilon_2) \cdots (1 - \alpha)(1 - h(\alpha)).$$

The solution is to set $\epsilon_{t-i} = (1 - \alpha - z_i/Z_{i-1})(1 - 2\alpha)$, where $Z_0 = 1 - h(\alpha)$, and $Z_i = \text{root}\{h(z/Z_{i-1}) - z - h(\alpha)\}$.

**Corollary:** For a binary WOM with binary symmetric input noise,

$$C(t) = (1 - \beta)\left(\frac{1}{t} \log_2 \left(1 + 2^{h(\alpha)} - 1 - 2^{h(\alpha)} - 1\right) - h(\alpha)\right).$$

Proving this result involves finding $\epsilon_1, \epsilon_2, \cdots, \epsilon_{t-1}$, which maximize

$$(h(\alpha * \epsilon_1) - h(\alpha)) + (1 - \alpha * \epsilon_1)(h(\alpha * \epsilon_2) - h(\alpha)) + \cdots + (1 - \alpha * \epsilon_1)(1 - \alpha * \epsilon_2) \cdots (1 - \alpha * \epsilon_{t-1})(1 - h(\alpha)).$$

The solution is

$$\epsilon_{t-i} = \frac{1 - \alpha(1 + \gamma_i)}{(1 - 2\alpha)(1 + \gamma_i)},$$

where

$$\gamma_i = \left(1 + \frac{1 - 2^{(h(\alpha)}}{1 - 2^{h(\alpha)}}\right)2^{-ih(\alpha)}.$$

**ACKNOWLEDGMENT**

The author would like to thank the reviewers for their useful comments, especially the reviewer who suggested Theorem 3.
where
\[ q_j = P(Y_j = 1|S_j = 0) \]
\[ F(q) = \sum_{j=1}^{n} P(S_j = 0)\left[h(q_j) - h(a)\right] + \theta_n \]
and where the maximum is over all \( q_j \) satisfying
\[ G(q) = \sum_{j=1}^{n} P(S_j = 0)q_j. \]
Using a Lagrange multiplier \( \lambda \), take the partial derivatives
\[
\frac{\partial [F(q) + \lambda G(q)]}{\partial q_j} = p(S_j = 0)\left[\log\left(\frac{1 - q_j}{q_j}\right) + \lambda\right].
\]
To set the derivatives to zero and to satisfy the constraint, we get
\[ q_j = \alpha \pm \epsilon \quad \text{or} \quad P(X_i = 1|S_j = 0) = \epsilon \]
independent of both the index \( j \) and the marginals \( P(S_j) \). Thus we obtain the bound
\[ R \leq (1 - E[\|\mathcal{Y}\|])\left(h(\alpha \pm \epsilon) - h(a)\right) + \theta_n. \]
This holds for every \( 1 \leq i \leq \ell \); thus from (5.3) we see that
\[ R_i \leq (1 - \beta)(1 - \alpha \pm \epsilon_1) \cdots (1 - \alpha \pm \epsilon_{\ell-1}) \cdot (h(\alpha \pm \epsilon_\ell) - h(a)) + \theta_n \]
in accordance with Theorem 4. Finally, we note that it is always sufficient to choose \( 0 \leq \epsilon_i \leq 1/2 \) for \( 1 \leq i \leq \ell - 1 \) and \( \epsilon_\ell = 1/2 \).

**References**


**Exponential Error Bounds for Random Codes in the Arbitrarily Varying Channel**

THOMAS ERICSON, MEMBER, IEEE

Abstract—Random codes for the arbitrarily varying channel are investigated. The code ensemble is restricted to increase only exponentially with codeword length; a quantity called key rate is used as a measure of the rate of increase. The reliability function for such codes is related to the reliability function for codes with unlimited key rate. Explicit results are obtained in some examples of practical interest.

I. INTRODUCTION

**STATEMENT**

Let \( \mathcal{X} \) and \( \mathcal{Y} \) be two finite sets, and let \( \mathcal{F} \) be an arbitrary set that is not necessarily finite. Let \( W : \mathcal{X} \times \mathcal{F} \rightarrow \mathcal{Y} \) be a probability matrix:

\[
W(y|x,s) \geq 0; \quad (x,s,y) \in \mathcal{X} \times \mathcal{F} \times \mathcal{Y}
\]

\[
\sum_{y \in \mathcal{Y}} W(y|x,s) = 1; \quad (x,s) \in \mathcal{X} \times \mathcal{F}
\]

and denote by \( \mathcal{V}^n \) the set of all probability matrices \( V_n : \mathcal{X}^n \rightarrow \mathcal{Y}^n \) of the form

\[
V_n(y|x,s) = \prod_{i=1}^{n} W(y|x_i,s_i)
\]

for some \( s = s_1s_2 \cdots s_n \in \mathcal{F}^n \). The sequence \( \{ V_n \}_{n=1}^{\infty} \) is called an arbitrarily varying channel (AVC) [1, p. 204]. The matrix \( W \) will be called the kernel. The interpretation is as follows.

A legal user tries to transmit a message \( m \in \mathcal{M}_s = \{1, 2, \cdots, M_s\} \) with the aid of a sequence \( x = x_1x_2 \cdots x_n \)