

Existence, Uniqueness, and Optimality of Sibling-Property Codes for Infinite Sources

Matthew Klimesh

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA, USA

Email: matthew.a.klimesh@jpl.nasa.gov

Robert J. McEliece

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA, USA

Email: rjm@systems.caltech.edu

Abstract—By definition Huffman codes only exist for finite sources, since the Huffman algorithm cannot be applied to an infinite source. On the other hand, Gallager’s sibling property, which was introduced as a characterization of Huffman codes, extends naturally to (countably) infinite sources. Thus we define a Huffman-Gallager code as any code that has the sibling property, and we present some basic facts about such codes: (1) For any source, a Huffman-Gallager code exists and its list of node probabilities is unique. (2) A Huffman-Gallager code is optimal, and given an optimal code, there exists a Huffman-Gallager code with the same codeword lengths. (3) For sources with infinite entropy, the existence and uniqueness results continue to hold, and the optimality results hold for a natural extended form of optimality.

I. INTRODUCTION

The Huffman algorithm [1] provides a simple method of constructing a variable length, prefix-condition code for a source with a finite alphabet, such that the code is optimal in the sense that the expected encoded length is minimized. Here and throughout this paper a source is assumed to be memoryless and have known symbol probabilities. A Huffman code for a source is defined as a code that can be generated by applying the Huffman algorithm to the source. Although a Huffman code is optimal, in general there are additional optimal codes that are not Huffman codes.

Gallager [2] has shown that a prefix-condition code for a source with a finite alphabet is a Huffman code if and only if it has the *sibling property*. Roughly speaking, the sibling property holds for a binary prefix-condition code if in the tree representation of the code, every node except the root has a sibling, and for any two pairs of siblings, the node probabilities for one of the pairs are each greater than or equal to the node probabilities of the other pair.

The sibling property provides a way to extend the concept of a Huffman code to countably infinite sources. We define a *Huffman-Gallager* code to be a prefix-condition code that has the sibling property, whether the corresponding source alphabet is finite or infinite. It is relatively straightforward to show that a code for a source with an infinite alphabet (but finite entropy) is optimal if it has the

sibling property (see Theorem 3 in Section III below). The optimality of sibling-property codes for infinite sources does not seem to have been explicitly noted previously, but an optimality-establishing technique that is essentially equivalent to checking that the sibling property holds has been used by Gallager [3] and later by others [4]–[7], and a closely related technique was used in [8]. These techniques involve forming a sequence of finite “reduced” sources by combining symbol probabilities from the original source, exhibiting Huffman codes for these sources, and examining the limit as the alphabet sizes of these sources become large.

Apart from explicitly noting that the sibling property implies optimality, our aim here is to demonstrate several basic properties of Huffman-Gallager codes. For simplicity we restrict our attention to binary codes.

In Section IV, we prove an existence result and a uniqueness result for Huffman-Gallager codes. They can be summarized roughly as follows: for any source, a Huffman-Gallager code exists, and any two such codes have the same multiset of node probabilities in their tree representations.

The definition of a Huffman-Gallager code makes sense (and our existence and uniqueness results apply) whether the entropy of the source is finite or infinite. Thus it is natural to desire to formulate an optimality result that holds in either case. Toward this end, in Section V we use two closely related definitions of optimality that each apply in either case. These definitions can be shown to be equivalent. One of these definitions was introduced by Kato et al. [8]. Both coincide with the usual definition of optimality when the source has finite entropy. We show that a Huffman-Gallager code for a source is optimal in the extended sense. We also show that if a given code is optimal in the extended sense for a source, then there exists a Huffman-Gallager code for the source such that for each source symbol, the codeword length for the symbol in the Huffman-Gallager code is the same as in the given code.

Our main motivation for considering codes for sources with infinite entropy is to give a more complete mathematical characterization of the properties of Huffman-Gallager codes. In practice one does not expect to be tasked with

encoding for a source with infinite entropy. However, we argue that consideration of infinite-entropy sources is not completely frivolous from a practical standpoint, because it could be reasonable to model a source as having an infinite entropy distribution. After all, models in which a source has an infinite alphabet have been given much attention and are frequently used, despite the fact that such sources are arguably always idealized models of finite sources.

II. PRELIMINARIES

Given a probability distribution P , the probability corresponding to source symbol i is denoted $P(i)$. We assume all source symbol probabilities are nonzero. For our purposes a code is a mapping from a source symbol set to a collection of binary codewords such that the prefix condition is satisfied. (As is well-known, the Kraft inequality [9] together with the McMillan inequality [10], along with their extensions to infinite alphabets [11], imply that there is no need to consider codes that do not satisfy the prefix condition, at least in the settings considered here.) Given a code C , the codeword for source symbol i is denoted $C(i)$. A length function L is a mapping from a source alphabet X to the positive integers such that the Kraft inequality, $\sum_{i \in X} 2^{-L(i)} \leq 1$, is satisfied. Any code has a corresponding length function, defined in the obvious way from the codeword lengths.

The average codeword length corresponding to length function L and probability distribution P defined on source alphabet X is denoted $\ell(L, P)$ and is given by $\ell(L, P) = \sum_{i \in X} P(i)L(i)$. If a code C has length function L , the average codeword length obtained by applying C to a source with the distribution P is denoted $\ell(C, P)$ and is given by $\ell(C, P) = \ell(L, P)$. Note that the source alphabet is implicit here, and for brevity we say “code C for distribution P ” as shorthand for “code C for a source with distribution P .”

For a probability distribution with finite entropy, an optimal code (or an optimal length function) is one that minimizes the average codeword length. Because we have defined length functions to satisfy the Kraft inequality, for any length function L there exists a code with L as its length function. It is clear that a code is optimal if and only if its length function is optimal. It is known that for any distribution with finite entropy, an optimal code exists [12]. We denote the minimum average codeword length by $\ell(P)$.

Any prefix-condition binary code can be represented by a binary tree in which left-branches are labeled with zeros, right-branches are labeled with ones, the leaf nodes are labeled with source symbols, and the sequence of branch labels on the path from the root to a leaf form the codeword for the leaf’s symbol. For our purposes, a code and its corresponding tree are the same object and we frequently emphasize the tree viewpoint.

Given a binary string s , we use $s0$ and $s1$ to represent the string concatenated with a 0 and a 1, respectively. Any node in the tree representation of a code can be identified by the binary string corresponding to the path from the root to the node. This string is a prefix of the string for any descendant of the node. A pair of nodes are siblings if and only if they are identified by strings of the form $s0$ and $s1$ for some s . We use “node s ” as a shorthand for “node identified with string s .”

Given a code C for source alphabet X and a probability distribution P on X , we associate probabilities with the nodes of the tree representation of C as follows. The probability corresponding to the leaf node labeled with source symbol i is $P(i)$. The probability corresponding to an internal node is the sum, over all of the node’s descendant leafs, of the probabilities corresponding to these leafs. Equivalently, the probability corresponding to an internal node can be defined recursively as the sum of the probabilities corresponding to the node’s children. For brevity, we refer to a probability corresponding to a node as the “node’s probability” or the “probability of the node.” We note that given a binary string s corresponding to a node of a code C , the probability of node s with respect to a distribution P is equal to the probability that s is a prefix of, or is equal to, the codeword resulting from choosing a symbol chosen according to P and encoding it with C .

The definition of a Huffman-Gallager code can be formalized as follows:

Definition: A code C for a distribution P has the sibling property if every node except the root node has a sibling, and for any two sibling pairs $(s0, s1)$ and $(t0, t1)$, the corresponding node probability pairs $(\alpha_{s0}, \alpha_{s1})$ and $(\alpha_{t0}, \alpha_{t1})$ satisfy

$$\begin{aligned} \max(\alpha_{s0}, \alpha_{s1}) &\leq \min(\alpha_{t0}, \alpha_{t1}) && \text{or} \\ \min(\alpha_{s0}, \alpha_{s1}) &\geq \max(\alpha_{t0}, \alpha_{t1}). \end{aligned} \quad (1)$$

A code with the sibling property is called a Huffman-Gallager code. We refer to the condition (1) as the sibling-property inequality.

Remark. A code in which each non-root node has a sibling is said to have a full encoding tree. A code that satisfies the Kraft inequality with equality is said to be exhaustive. For finite codes, having a full encoding tree is equivalent to being exhaustive. However, as pointed out in [12], when dealing with infinite sources, a code can have a full encoding tree without being exhaustive. The definition of a Huffman-Gallager code contains the weaker requirement, namely that of having a full encoding tree. However we will see later that a Huffman-Gallager code must be exhaustive.

We will frequently want to form a finite source from an infinite source by partitioning the source symbols into a finite number of subsets. A finite partition $\mathcal{A} = \{A(1), \dots, A(n)\}$ of the source symbols for a distribution P on a possibly infinite alphabet induces a distribution

$P_{\mathcal{A}}$ on the symbols $\{1, \dots, n\}$ with probabilities given by $P_{\mathcal{A}}(i) = \sum_{j \in A(i)} P(j)$. For some partitions a code for the original source has a clear counterpart for the induced source; in this regard the following concept (from [8]) is useful:

Definition: A finite partition $\mathcal{A} = \{A(1), \dots, A(n)\}$ of a source is *code-compatible* with a code C for the source if for each $i \in \{1, \dots, n\}$, there is a node s_i in the tree for C such that either $A(i)$ is exactly equal to the set of symbols that correspond to leaves that are descendants of s_i , or s_i is a leaf and $A(i)$ is equal to the singleton set containing the symbol corresponding to s_i . The code (often denoted $C_{\mathcal{A}}$) corresponding to the tree formed by replacing each s_i with a leaf labeled with source symbol i is said to be induced by the partition.

A similar concept applies to length functions:

Definition: Given a length function L that is exhaustive (its Kraft sum $\sum_{j \in X} 2^{-L(j)}$ is equal to 1), we say that a finite partition $\mathcal{A} = \{A(1), \dots, A(n)\}$ of X is *length-compatible* with L if there is a (necessarily exhaustive) length function $L_{\mathcal{A}}$ for which $2^{-L_{\mathcal{A}}(i)} = \sum_{j \in A(i)} 2^{-L(j)}$ for each i . We say $L_{\mathcal{A}}$ is induced by the partition.

The following propositions are straightforward.

Proposition 1: Suppose \mathcal{A} is a finite partition that is code-compatible with a code C , and suppose s is a binary string that corresponds to a node in the induced code $C_{\mathcal{A}}$. Then s also corresponds to a node in C , and the probability of node s in C with respect to a source distribution P is the same as the probability of node s in $C_{\mathcal{A}}$ with respect to the induced distribution $P_{\mathcal{A}}$. Also, the node s has a sibling in $C_{\mathcal{A}}$ if and only if the node s has a sibling in C . Finally, if C is a Huffman-Gallager code for P , then $C_{\mathcal{A}}$ is a Huffman code for $P_{\mathcal{A}}$.

Proposition 2: Given a code C and a finite set S of binary strings corresponding to nodes of C , there exists a finite partition \mathcal{A} that is code-compatible with C such that each $s \in S$ corresponds to a node in the induced code $C_{\mathcal{A}}$. In particular, given a finite set of source symbols, there exists a finite partition \mathcal{A} , code-compatible with C , for which $\{i\} \in \mathcal{A}$ for each i in the set.

III. OPTIMALITY FOR FINITE-ENTROPY SOURCES

Although our result on the optimality of Huffman-Gallager codes for any source (Theorem 9) includes the case where the source has finite entropy (as the usual and extended definitions of optimality coincide for such sources), the finite-entropy case is simpler, and may be of more interest, so we present it on its own.

Theorem 3: A Huffman-Gallager code for a finite-entropy source is optimal.

This result can be obtained immediately from Theorem 2 in [8] (along with our Proposition 1), but we include a sketch of a proof. The underlying reasoning is not very different from that used in the optimality results in [4]–[8].

Sketch of proof. Suppose C is a Huffman-Gallager code for a source with distribution P . If the source has a

finite alphabet, then C is a Huffman code for P , so the desired optimality follows. Thus we suppose the source has a countably infinite alphabet, which we take to be the positive integers \mathbf{N} . Let $\mathcal{A}_1, \mathcal{A}_2, \dots$ be a sequence of finite partitions that are code-compatible with C such that for each source symbol i , the set $\{i\}$ is a member of \mathcal{A}_n for $n \geq i$. Such a sequence can be easily constructed, for example with Proposition 2.

By Proposition 1, each induced code $C_{\mathcal{A}_n}$ is a Huffman code for the induced distribution $P_{\mathcal{A}_n}$, so $C_{\mathcal{A}_n}$ is optimal for $P_{\mathcal{A}_n}$. For $n \geq 1$, we then can deduce that

$$\ell(C_{\mathcal{A}_n}, P_{\mathcal{A}_n}) = \ell(P_{\mathcal{A}_n}) \leq \ell(P) \leq \ell(C, P). \quad (2)$$

We note that it is well-known (and easily shown) that $\ell(P)$ is finite when the source has finite entropy.

It can be shown that $\lim_{n \rightarrow \infty} \ell(C_{\mathcal{A}_n}, P_{\mathcal{A}_n}) = \ell(C, P)$. From this fact the desired conclusion $\ell(C, P) = \ell(P)$ is reached by letting $n \rightarrow \infty$ in (2). ■

IV. EXISTENCE AND UNIQUENESS

In this section we present our existence and uniqueness results. These results apply whether or not the source has finite entropy. We note that both results are straightforward for sources with finite alphabets.

The existence result is straightforward to state:

Theorem 4: A Huffman-Gallager code exists for any source.

Given a Huffman-Gallager code for a source, there is a corresponding multiset consisting of the pairs of probabilities of sibling pairs in the code tree. With this notion our uniqueness result is the following:

Theorem 5: Any two Huffman-Gallager codes for the same source have identical corresponding multisets of sibling pair probabilities.

The conclusion of Theorem 5 is equivalent to the assertion that if we list, in decreasing order, the node probabilities of a Huffman-Gallager code for a source, then the resulting list will be identical to the corresponding list for any other Huffman-Gallager code for the same source.

Our proof of Theorem 4 uses the following lemma. It follows from a more precise result in [13].

Lemma 6: For any nonzero probability value p , there exists an integer $\eta(p)$ such that if C is a Huffman code for a finite probability distribution P , and $P(i) = p$ for some symbol i , then the length of $C(i)$ is at most $\eta(p)$.

This lemma essentially says that given a symbol probability, there are a limited number of possibilities for the length of the codeword for that symbol in a Huffman code.

Sketch of proof of Theorem 4. For a source with a finite alphabet, this result follows from Gallager [2] and Huffman [1]. Thus suppose P is a probability distribution on an infinite set, which we take to be \mathbf{N} . For $j \geq 1$ let P_j be the distribution on the set of $j+1$ symbols $\{1, \dots, j\} \cup \{\infty\}$ with probabilities given by $P_j(i) = P(i)$ when $1 \leq i \leq j$, and $P_j(\infty) = \sum_{i > j} P(i)$. Let C_j be a Huffman code for P_j .

We next will define an infinite sequence of sets S_0, S_1, \dots with each $S_n \subset \mathbf{N}$ and an infinite sequence of codewords $C^*(1), C^*(2), \dots$. The members of S_n can be regarded as indices specifying a subsequence of $\{C_i\}_{i=1}^\infty$. The codewords $C^*(1), C^*(2), \dots$ will collectively form the desired Huffman-Gallager code for the distribution P .

The sequences S_0, S_1, \dots and $C^*(1), C^*(2), \dots$ are defined recursively and will satisfy the following properties:

- 1) If $n \geq 1$ then $S_n \subset S_{n-1}$.
- 2) Each S_n is infinite.
- 3) Each S_n satisfies $\min S_n > n$.
- 4) If $i \in S_n$ then $C_i(j) = C^*(j)$ for $j = 1, \dots, n$.

The recursive definition is as follows. Let $S_0 = \mathbf{N}$. If $n \geq 1$, then the definition of S_n relies on S_{n-1} and $C^*(1), \dots, C^*(n-1)$ satisfying the properties above. Observe that if $i \in S_{n-1}$, then $n \leq i$ and so $P_i(n) = P(n)$. Therefore, by Lemma 6 the codewords $C_i(n)$, as i runs through the elements of S_{n-1} , are bounded in length. Thus there are a finite number of different codewords $C_i(n)$ as i runs through the elements of S_{n-1} . We may therefore define $C^*(n)$ to be a codeword that occurs infinitely often in the sequence $\{C_i(n)\}_{i \in S_{n-1}}$. (For definiteness, we can, for example, define $C^*(n)$ to be the lexicographically earliest codeword among those that occur infinitely often.) Finally, let $S_n = \{i \in S_{n-1} \mid i > n \text{ and } C_i(n) = C^*(n)\}$. It is easily checked that the above properties 1-4 are satisfied for this S_n , so our recursive definition is valid.

The codewords $C^*(1), C^*(2), \dots$ collectively form a code C^* . We claim that C^* is a Huffman-Gallager code for P . For any $n > 0$ the set of codewords $\{C^*(1), \dots, C^*(n)\}$ satisfies the prefix condition because for any $i \in S_n$ they are codewords in the Huffman code C_i . Thus the whole code C^* must be a prefix condition code. The remaining requirements, namely that every non-root node of C^* has a sibling and that the sibling-property inequality holds, are more involved to show and we omit the details. ■

The following lemma is used in our proof of Theorem 5.

Lemma 7: Suppose P_V and P_W are probability distributions on an alphabet X of size n . For $Z \in \{V, W\}$ let $\alpha_Z(1), \dots, \alpha_Z(2n-2)$ be the (non-root) node probabilities in a Huffman code for P_Z , listed in decreasing order. Let $\Delta = \sum_{i \in X} |P_V(i) - P_W(i)|$. Then $|\alpha_V(j) - \alpha_W(j)| \leq \Delta$ for $j \in \{1, \dots, 2n-2\}$.

Sketch of Proof. We take the source alphabet X to be $\{1, \dots, n\}$. All non-root nodes of a Huffman code occur as sibling pairs and corresponds to probabilities that are merged in the Huffman algorithm. Based on this idea, for $Z \in \{V, W\}$ the probabilities of the nodes of a code generated by the Huffman algorithm for P_Z can be computed as follows. Let $x_Z^{(1)}(i) = P_Z(i)$ for $i \in \{1, \dots, n\}$. For $k \in \{1, \dots, n-1\}$ define $y_Z^{(k)}(1), \dots, y_Z^{(k)}(n-k+1)$ as the values $x_Z^{(k)}(1), \dots, x_Z^{(k)}(n-k+1)$ sorted into decreasing order. Define $x_Z^{(k+1)}(1), \dots, x_Z^{(k+1)}(n-k)$ from $y_Z^{(k)}(1), \dots, y_Z^{(k)}(n-k+1)$ by merging the last two values; that is, $x_Z^{(k+1)}(i) = y_Z^{(k)}(i)$ when $i \in \{1, \dots, n-k\}$, and

$x_Z^{(k+1)}(n-k+1) = y_Z^{(k)}(n-k+1) + y_Z^{(k)}(n-k+2)$. With these definitions, the probabilities of the nodes of a Huffman code are the last two values of $y_Z^{(k)}$ for $k = 1, \dots, n-1$. Specifically, in descending order these probabilities are $y_Z^{(n-1)}(1), y_Z^{(n-1)}(2), y_Z^{(n-2)}(2), y_Z^{(n-2)}(3), \dots, y_Z^{(1)}(n-1), y_Z^{(1)}(n)$.

It can be shown that if $\sum_{i=1}^{n-k+1} |x_V^{(k)}(i) - x_W^{(k)}(i)| \leq \Delta$ for some k , then $\sum_{i=1}^{n-k+1} |y_V^{(k)}(i) - y_W^{(k)}(i)| \leq \Delta$. The triangle inequality implies that if $\sum_{i=1}^{n-k+1} |y_V^{(k)}(i) - y_W^{(k)}(i)| \leq \Delta$ then $\sum_{i=1}^{n-k} |x_V^{(k+1)}(i) - x_W^{(k+1)}(i)| \leq \Delta$. By hypothesis, $\sum_{i=1}^n |x_V^{(1)}(i) - x_W^{(1)}(i)| \leq \Delta$. Therefore, by induction, we must have $\sum_{i=1}^{n-k+1} |y_V^{(k)}(i) - y_W^{(k)}(i)| \leq \Delta$ for $k = 1, \dots, n-1$.

The conclusion is now straightforward since $|\alpha_V(j) - \alpha_W(j)|$ is equal to one of the last two terms of the sum $\sum_{i=1}^{n-k+1} |y_V^{(k)}(i) - y_W^{(k)}(i)|$ for $k = n - \lceil j/2 \rceil$. ■

Sketch of proof of Theorem 5. For distributions on finite alphabets, the conclusion is clear, following easily from either the Huffman algorithm itself, or alternatively from Lemma 7. Thus we suppose C_V and C_W are Huffman-Gallager codes for a distribution P on an infinite alphabet, which we take to be \mathbf{N} . For $Z \in \{V, W\}$ let $\{\alpha_Z(i)\}_{i=1}^\infty$ be the node probabilities in C_Z with respect to P , listed in decreasing order.

Let j be an arbitrary positive integer and let $\epsilon > 0$ be arbitrary. We will show that $|\alpha_V(j) - \alpha_W(j)| \leq \epsilon$. Because j and $\epsilon > 0$ are arbitrary, we will be able to conclude that $\alpha_V(j) = \alpha_W(j)$ for all $j \in \mathbf{N}$, and the proof will be complete.

Choose k so that $\sum_{i>k} P(i) < \epsilon/2$. For $Z \in \{V, W\}$ let S_Z be a set of binary strings that is the union of (a) a set of strings identifying j distinct nodes of C_Z that have associated probabilities $\alpha_Z(1), \dots, \alpha_Z(j)$, and (b) the set of strings $C_Z(1), \dots, C_Z(k)$. By Proposition 2, for $Z \in \{V, W\}$ there exists a finite partition \mathcal{A}_Z that is code-compatible with C_Z such that each $s \in S_Z$ corresponds to a node in the induced code. Note that any such partition can be increased in size by one by splitting a set corresponding to an internal node of C_Z into the two sets corresponding to the node's children. Thus we may assume that $|\mathcal{A}_V| = |\mathcal{A}_W| = n$ for some n . We denote the contents of these sets according to $\mathcal{A}_V = \{A_V(1), \dots, A_V(n)\}$ and $\mathcal{A}_W = \{A_W(1), \dots, A_W(n)\}$. By assumption $\{i\}$ is a member of \mathcal{A}_V and \mathcal{A}_W for $i = 1, \dots, k$, so we may assume that $A_V(i) = A_W(i) = \{i\}$ for $i = 1, \dots, k$.

For notational conciseness, we temporarily depart from our usual notation for induced codes and distributions. Denote by C'_V the code induced by \mathcal{A}_V on C_V , and by C'_W the code induced by \mathcal{A}_W on C_W . Denote the probability distributions induced by \mathcal{A}_V and \mathcal{A}_W on P by P'_V and P'_W , respectively.

For $Z \in \{V, W\}$ let $\alpha'_Z(i)$ and be the i th value in the decreasing list of probabilities of nodes of C'_Z . By Proposition 1 and our conditions on \mathcal{A}_Z , we see that the values $\alpha'_Z(1), \dots, \alpha'_Z(j)$ must be the same as $\alpha_Z(1), \dots, \alpha_Z(j)$. In

particular, we have $\alpha'_V(j) = \alpha_V(j)$ and $\alpha'_W(j) = \alpha_W(j)$.

Note that $P'_V(i) = P'_W(i) = P(i)$ for $i = 1, \dots, k$. Therefore

$$\sum_{i=1}^n |P'_V(i) - P'_W(i)| = \sum_{i=k+1}^n |P'_V(i) - P'_W(i)| < \epsilon,$$

where the final inequality follows from our choice of k . Thus by Lemma 7, we must have $|\alpha'_V(j) - \alpha'_W(j)| \leq \epsilon$. Therefore $|\alpha_V(j) - \alpha_W(j)| \leq \epsilon$, as desired. ■

V. EXTENDED OPTIMALITY

We now extend the notion of optimality to sources with infinite entropy, with the motivation of showing that a Huffman-Gallager code for such a source is optimal in the extended sense. One of these notions applies to codes directly, and the other applies to length functions. The first of these was put forth by Kato et al. [8].

Definition: A code C is optimal in the extended sense for a probability distribution P if for any finite code-compatible partition \mathcal{A} of the source, the induced code $C_{\mathcal{A}}$ is optimal (in the usual finite sense) for the induced source distribution $P_{\mathcal{A}}$.

This extended optimality definition is convenient to work with, and can be argued to be intuitively reasonable. However, under this extension it may not be immediately clear that a code with the same length function as an optimal code is also optimal, and we would like the optimality status of a code to depend only on its length function. Thus we introduce the following definition.

Definition: A length function L is optimal in the extended sense for a given probability distribution P if it is exhaustive (has Kraft sum equal to 1), and for any finite partition \mathcal{A} of the source that is length-compatible with L , the induced length function $L_{\mathcal{A}}$ is optimal for the induced source distribution $P_{\mathcal{A}}$.

Fortunately, these definitions turn out to be closely related:

Theorem 8: Suppose a code C has length function L . Then C is optimal in the extended sense for a probability distribution P if and only if L is optimal in the extended sense for P .

The proof is omitted here.

For a source with finite entropy, it follows from Kato et al. [8] that the usual definition of an optimal code coincides with the extended optimality version. Therefore, for such sources Theorem 8 implies that the usual definition of an optimal length function coincides with our length function extended optimality definition.

We now link our notions of optimality with Huffman-Gallager codes.

Theorem 9: A Huffman-Gallager code for a source is optimal in the extended sense for the source (and its length function is likewise optimal in the extended sense for the source).

Proof. This follows immediately from the extended definition of optimality of a code, and from the fact that

a finite code-compatible partition of a Huffman-Gallager code induces a Huffman code (Proposition 1). The optimality of the length function follows from Theorem 8. ■

Since a Huffman-Gallager code has an extended-sense optimal length function, and an extended-sense optimal length function is by definition exhaustive, any Huffman-Gallager code must be exhaustive.

Finally, we present a result that shows that a code that is optimal but not a Huffman-Gallager code is nonetheless closely related to a Huffman-Gallager code:

Theorem 10: If a length function L is optimal in the extended sense for a source distribution P , then there exists a Huffman-Gallager code for P with length function L .

The proof is omitted here.

VI. CONCLUSION

Gallager's sibling property provides an extension of the concept of a Huffman code to infinite sources. Here we have presented some properties of Huffman-Gallager codes: existence, uniqueness, and optimality (usual and extended). We hope that this serves to establish some of the basic mathematical structure of optimal prefix-condition codes.

REFERENCES

- [1] D. A. Huffman, "A method for the construction of minimum redundancy codes," *Proc. IRE*, vol. 40, no. 9, pp. 1098–1101, Sept. 1952.
- [2] R. G. Gallager, "Variations on a theme by Huffman," *IEEE Trans. Inform. Theory*, vol. IT-24, no. 6, pp. 668–674, Nov. 1978.
- [3] R. G. Gallager and D. C. Van Voorhis, "Optimal source codes for geometrically distributed integer alphabets," *IEEE Trans. Inform. Theory*, vol. IT-21, no. 2, pp. 228–230, Mar. 1975.
- [4] P. Humblet, "Optimal source coding for a class of integer alphabets," *IEEE Trans. Inform. Theory*, vol. IT-24, no. 1, pp. 110–112, Jan. 1978.
- [5] J. Abrahams, "Huffman-type codes for infinite source distributions," *J. Franklin Inst.*, vol. 331B, no. 3, pp. 265–271, May 1994.
- [6] N. Merhav, G. Seroussi, and M. J. Weinberger, "Optimal prefix codes for sources with two-sided geometric distributions," *IEEE Trans. Inform. Theory*, vol. 46, no. 1, pp. 121–135, Jan. 2000.
- [7] F. Bassino, J. Clément, G. Seroussi, and A. Viola, "Optimal prefix codes for some families of two-dimensional geometric distributions," in *Proc. 2006 Data Compression Conf. (DCC 2006)*, Snowbird, Utah, USA, Mar. 28–30, 2006, pp. 113–122.
- [8] A. Kato, T. S. Han, and H. Nagaoka, "Huffman coding with an infinite alphabet," *IEEE Trans. Inform. Theory*, vol. 42, no. 3, pp. 977–984, May 1996.
- [9] L. G. Kraft, "A device for quantizing, grouping, and coding amplitude modulated pulses," Master's thesis, MIT, Cambridge, MA, 1949.
- [10] B. McMillan, "Two inequalities implied by unique decipherability," *IRE Trans. Inform. Theory*, vol. 2, no. 4, pp. 115–116, Dec. 1956.
- [11] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
- [12] T. Linder, V. Tarokh, and K. Zeger, "Existence of optimal prefix codes for infinite source alphabets," *IEEE Trans. Inform. Theory*, vol. 43, no. 6, pp. 2026–2028, Nov. 1997.
- [13] G. Katona and O. Nemetz, "Huffman codes and self-information," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 3, pp. 337–340, May 1976.