Tight Upper Bounds on the Redundancy of Optimal Binary AIFV Codes

Weihua Hu
Dept. of Mathematical Eng.
The University of Tokyo, Japan
Email: weihua916@gmail.com

Hirosuke Yamamoto
Dept. of Complexity Sci. and Eng.
The University of Tokyo, Japan
Email: Hirosuke@ieee.org

Junya Honda
Dept. of Complexity Sci. and Eng.
The University of Tokyo, Japan
Email: honda@it.k.u-tokyo.ac.jp

***Abstract*—AIFV codes are lossless codes that generalize the class of instantaneous FV codes. The code uses multiple code trees and assigns source symbols to incomplete internal nodes as well as to leaves. AIFV codes are empirically shown to attain better compression ratio than Huffman codes. Nevertheless, an upper bound on the redundancy of optimal binary AIFV codes is only known to be 1, the same as the bound of Huffman codes. In this paper, the upper bound is improved to 1/2, which is shown to be tight. Along with this, a tight upper bound on the redundancy of optimal binary AIFV codes is derived for the case $p_{\max}$ ≥1/2, where $p_{\max}$ is the probability of the most likely source symbol. This is the first theoretical work on the redundancy of optimal binary AIFV codes, suggesting superiority of the codes over Huffman codes.**

I. Introduction

Fixed-to-Variable length (FV) codes map source symbols to variable length codewords, and can be represented by code trees. In the case of a binary instantaneous FV code, source symbols are assigned to leaves of the binary tree. The codeword for each source symbol is then given by the path from the root to the corresponding leaf. It is well-known by McMillan's paper [1] that Huffman codes [2] attain the minimum average code length in the class of uniquely decodable FV codes. However, it was assumed in [1] that a single code tree is used for a uniquely decodable FV code. Hence, if we use multiple code trees for a uniquely decodable FV code, it may be possible to attain better compression rate than Huffman codes.

Recently, Almost Instantaneous Fixed-to-Variable length (AIFV) codes were proposed as a new class of uniquely decodable codes that generalize the class of instantaneous FV codes [3][4]. Unlike an instantaneous FV code, which uses only one code tree, an AIFV code is allowed to use multiple code trees. Furthermore, source symbols on the AIFV code trees are assigned to incomplete internal nodes as well as to leaves. In the case of a binary AIFV code [4], two code trees are used in a way such that decoding delay is at most two bits, which is why the code is called "almost" instantaneous.

Binary AIFV codes are empirically shown to be powerful in data compression. Not only do the codes attain better compression ratio than Huffman codes, experiments suggest that for some sources, AIFV codes can even beat Huffman codes for $\mathcal{X}^2$, where $\mathcal{X}$ is the source alphabet [4]. Nonetheless, few theoretical results are known about the codes. In particular, an upper bound on the redundancy (the expected code length minus entropy) of optimal binary AIFV codes is only known to be 1, a trivial bound derived from the fact that binary AIFV codes include Huffman codes.

In this paper, we present the first non-trivial theoretical result on the redundancy of optimal binary AIFV codes, suggesting superiority of the codes over Huffman codes. In specific, we show that the tight upper bound on the redundancy of optimal binary AIFV codes is $\frac{1}{2}$, the same bound as that of Huffman codes for $\mathcal{X}^2$. Note that for $K = |\mathcal{X}|$, the size of memory required to store code trees is $\mathcal{O}(K)$ for a binary AIFV code, while $\mathcal{O}(K^2)$ for a Huffman code for $\mathcal{X}^2$ [4]. Thus, a binary AIFV code can attain competitive compression ratio of a Huffman code for $\mathcal{X}^2$ with much less memory requirement.

We also derive a tight upper bound on the redundancy of optimal binary AIFV codes in terms of $p_{\max} \geq \frac{1}{2}$, where $p_{\max}$ is the probability of the most likely source symbol. We compare this upper bound with its Huffman counterpart [5] and show that optimal binary AIFV codes significantly improve the bound for every $p_{\max} \geq \frac{1}{2}$.

II. Preliminaries

A. Binary AIFV codes

A binary AIFV code uses two binary code trees, denoted by T_0 and T_1, in a way such that the code is uniquely decodable and the decoding delay is at most two bits. To illustrate how this works, we begin with a list of properties satisfied by the trees of a binary AIFV code.

1) Incomplete internal nodes (nodes with one child) are classified into two categories, master nodes and slave nodes.
2) Source symbols are assigned to either master nodes or leaves.
3) The child of a master node must be a slave node, and the master node is connected to its grandchild by code symbols '00'.
4) The root of T_1 must have two children. The child connected by '0' from the root is a slave node. The slave node is connected by '1' to its child.

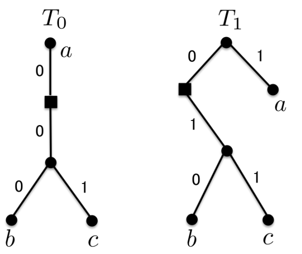


Fig. 1. An example of binary AIFV code trees.

Properties 1) and 2) indicate that a binary AIFV code allows source symbols to be assigned to the incomplete internal nodes. Properties 3) and 4) can be interpreted as constraints on the tree structures to ensure that decoding delay is at most two bits. Fig. 1 illustrates an example of a binary AIFV code for $\mathcal{X} = \{a, b, c\}$, where slave nodes are marked with squares. It is easy to see that the trees satisfy all the properties of a binary AIFV code.

Given a source sequence $x_1x_2x_3\cdots$, an encoding procedure of a binary AIFV code goes as follows.

1) Use T_0 to encode the initial source symbol x_1.
2) When x_i is encoded by a leaf (resp. a master node), then use T_0 (resp. T_1) to encode the next symbol x_{i+1}.

Using a binary AIFV code of Fig. 1, a source sequence '$aabac$' is encoded to '1000011', where the code trees are visited in the order of $T_0 \to T_1 \to T_0 \to T_0 \to T_1$.

A codeword sequence $y_1y_2y_3\cdots \in \{0, 1\}^*$ is decoded as follows.

1) Use T_0 to decode the initial source symbol x_1.
2) Trace the codeword sequence as long as possible from the root in the current code tree. Then, output the source symbol assigned to the reached master node or leaf.
3) If the reached node is a leaf (resp. a master node), then use T_0 (resp. T_1) to decode the next source symbol from the current position on the codeword sequence.

The decoding process is guaranteed to visit the code trees in the same order as the corresponding encoding process does [4]. The codeword sequence '1000011' is indeed decoded to the original source sequence '$aabac$', using a sequence of trees, T_0, T_1, T_0, T_0 and T_1, in this order. Note that since the first codeword '1' on the code sequence cannot be traced on T_0 of Fig. 1, we output 'a', which is assigned to the root of T_0. When all source symbols are assigned to leaves of T_0, a binary AIFV code reduces to an instantaneous FV code.

The following defines an average code length of a binary AIFV code, denoted by L_{AIFV}.

$$L_{\mathrm{AIFV}} = P(T_0)L_{T_0} + P(T_1)L_{T_1}, \tag{1}$$

where $P(T_0)$ (resp. $P(T_1)$) is a stationary probability of T_0 (resp. T_1), and L_{T_0} (resp. L_{T_1}) is the average code length of T_0 (resp. T_1).

Let L_{OPT} be the average code length of the optimal binary AIFV code for a given source. Then, the redundancy of optimal binary AIFV code denoted by r_{AIFV}, is defined as

$$r_{\mathrm{AIFV}} \equiv L_{\mathrm{OPT}} - H(X), \tag{2}$$

where X is a random variable corresponding to the source. It is shown in [4] how we can obtain optimal binary AIFV code trees for a given source.

B. Sibling property of Huffman codes

Sibling property was first introduced in [5] as a structural characterization of Huffman codes. Consider a K-ary source and let T_{Huffman} denote the corresponding Huffman tree. Let the weight of a leaf be defined as the probability of corresponding source symbol. Also, let the weight of an internal node be defined recursively as the sum of the probabilities of the children. There are $2K-2$ nodes (except the root) on T_{Huffman}. Let $q_1, q_2, \ldots, q_{2K-2}$ be the weights of the nodes sorted in a non-increasing order, so that $q_1 \geq q_2, \cdots \geq q_{2K-2}$. By a slight abuse of notations, we identify q_k with the corresponding node itself in the rest of the paper.

We state the sibling property of Huffman codes, which will play an important role in the later proofs of the redundancy of optimal binary AIFV codes.

Definition 1 (Sibling property). *A binary code tree has the sibling property if there exists a sequence of nodes $q_1, q_2, \ldots, q_{2K-2}$, such that for every $k \in \{1, \ldots, K-1\}$, q_{2k} and q_{2k-1} are sibling on the tree.*

Theorem 1 ([5, Theorem 1]). *A binary instantaneous code is a Huffman code iff the code tree has the sibling property.*

C. Redundancy upper bounds of Huffman codes

It is well-known that an upper bound on the redundancy of Huffman codes is 1. Meanwhile, a lot of studies have shown that a better bound on the redundancy can be obtained when a source satisfies some predefined conditions. One such condition concerns with the value of $p_{\max}$ [5]–[9], where $p_{\max}$ is the probability of the most likely source symbol. The following bound is proven by Gallager [5].

Theorem 2 ([5, Theorem 2]). *For $p_{\max} \geq \frac{1}{2}$, the redundancy of binary Huffman codes is upper bounded by $2 - p_{\max} - h(p_{\max})$, where $h(\cdot)$ is the binary entropy function.*

Note that the bound provided by Theorem 2 is tight in the sense that a source with probabilities $(p_{\max}, 1 - p_{\max} - \delta, \delta)$ satisfies the bound with equality in the limit of $\delta \to 0$.

In Fig. 2, we summarize the upper bound results for Huffman codes in terms of $p_{\max}$. The bound is shown to be tight for $p_{\max} \geq \frac{1}{6}$ in [10]. We see from Fig. 2 that the redundancy of Huffman codes approaches to 1 only when source probabilities are extremely biased.

III. Redundancy upper bounds of optimal binary AIFV codes

We have reviewed some upper bound results on the redundancy of Huffman codes. In this section, we derive their counterparts on binary AIFV codes, which directly suggests

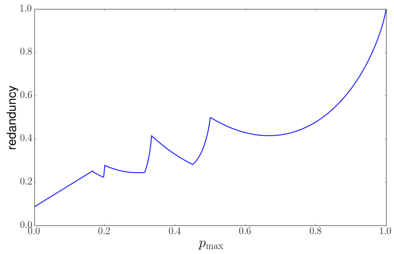


Fig. 2. The redundancy upper bounds of Huffman codes in terms of $p_{\max}$.

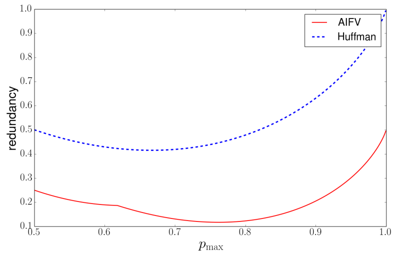


Fig. 3. Comparison on the redundancy upper bounds between binary AIFV codes and Huffman codes in terms of $p_{\max}$.

superiority of binary AIFV codes over Huffman codes in terms of compression ratio.

Theorem 3. *For $p_{\max} \geq \frac{1}{2}$, the redundancy of optimal binary AIFV codes is upper bounded by $f(p_{\max})$, where $f(x)$ is defined as follows.*

$$f(x) = \begin{cases} x^2 - 2x + 2 - h(x) & \text{if } \frac{1}{2} \leq x \leq \frac{-1+\sqrt{5}}{2}, \\ \frac{-2x^2+x+2}{1+x} - h(x) & \text{if } \frac{-1+\sqrt{5}}{2} \leq x < 1. \end{cases} \quad (3)$$

The above bound is tight in the sense that there exists a source distribution for any $\epsilon > 0$ such that the redundancy is larger than $f(p_{\max}) - \epsilon$.

Fig. 3 compares the upper bounds given by Theorems 2 and 3. We see that the upper bound on the redundancy of optimal binary AIFV codes is smaller than that of Huffman codes for every $p_{\max} \geq \frac{1}{2}$.

We also get Theorem 4, covering the case of $p_{\max} < \frac{1}{2}$.

Theorem 4. *For $p_{\max} < \frac{1}{2}$, the redundancy of optimal binary AIFV codes is at most $\frac{1}{4}$.*

The bound given by Theorem 4 is not necessarily tight. Yet the derived bound is sufficient to prove Corollary 1, which follows immediately from Theorems 3 and 4.

Corollary 1. *The redundancy of optimal binary AIFV codes is upper bounded by $\frac{1}{2}$ for any source.*

Note that the bound given by Corollary 1 is tight and is the same upper bound as that of Huffman codes for $\mathcal{X}^2$. In fact, it is empirically shown that for some sources, a binary AIFV code can beat a Huffman code for $\mathcal{X}^2$ [4]. Meanwhile, in terms of memory efficiency, a binary AIFV code only requires $\mathcal{O}(K)$ for storing code trees, while a Huffman code for $\mathcal{X}^2$ needs $\mathcal{O}(K^2)$. This suggests that a binary AIFV code is more memory efficient than a Huffman code for $\mathcal{X}^2$, while maintaining competitive compression performance against a Huffman code for $\mathcal{X}^2$.

IV. Proofs

We start with proving Theorem 3. The key to the proof is to transform a Huffman code tree into binary AIFV code trees. Then, we can utilize the structural property of the Huffman tree, namely the sibling property, in evaluating the redundancy of the binary AIFV code. For the notations on the sibling property, see Section II-B. We first prepare three lemmas, which provide useful inequalities for the later evaluation of the bounds.

Lemma 1. *Consider a Huffman tree and suppose that q_{2k-1} is not a leaf. Then, $q_{2k-1} \leq 2q_{2k}$ holds.*

Proof. Let q'_1 and q'_2 be children of q_{2k-1}. In the construction of the Huffman tree, q'_1 and q'_2 are merged before q_{2k-1} and q_{2k} are merged. Thus, $q'_1 \leq q_{2k}$ and $q'_2 \leq q_{2k}$ hold. Therefore, we get $q_{2k-1} = q'_1 + q'_2 \leq 2q_{2k}$. □

Lemma 2. *Assume $0 < 2w_2 < w_1$ and let $q \in \left[0, \frac{1}{2}\right]$ be arbitrary. Then,*

$$\begin{aligned} 2w_2 - (w_1 + w_2)h\left(\frac{w_2}{w_1 + w_2}\right) + qw_1 \\ < q\left(w_1 - w_2\right). \end{aligned} \quad (4)$$

Proof. Let $c \equiv \frac{w_1}{w_2} > 2$ and define $g(x) \equiv h(x) - 2x$. Subtracting the LHS of (4) from the RHS, we get

$$\begin{aligned} &-qw_2 - 2w_2 + (1+c)w_2 \cdot \left(g\left(\frac{1}{1+c}\right) + \frac{2}{1+c}\right) \\ &= w_2 \cdot \left(-q + (1+c) \cdot g\left(\frac{1}{1+c}\right)\right) \\ &\geq w_2 \cdot \left(-\frac{1}{2} + (1+c) \cdot g\left(\frac{1}{1+c}\right)\right) \\ &> 0. \end{aligned} \quad (5)$$

The last inequality follows from $\inf_{c>2}(1+c)g(\frac{1}{1+c}) = 0.754\cdots > \frac{1}{2}$. □

Lemma 3. *If $q_{2k-1} \leq 2q_{2k}$, then*

$$(q_{2k-1} + q_{2k})\left(1 - h\left(\frac{q_{2k}}{q_{2k-1} + q_{2k}}\right)\right) \leq \frac{1}{4}\left(q_{2k-1} - q_{2k}\right). \quad (6)$$

Proof. Since $q_{2k} \leq q_{2k-1} \leq 2q_{2k}$, it follows that $\frac{1}{3} \leq \frac{q_{2k}}{q_{2k-1}+q_{2k}} \leq \frac{1}{2}$. Further, since $\frac{1}{2}x + \frac{3}{4} \leq h(x)$ holds for $\frac{1}{3} \leq x \leq \frac{1}{2}$,

$$\begin{aligned} &(q_{2k-1} + q_{2k})\left(1 - h\left(\frac{q_{2k}}{q_{2k-1} + q_{2k}}\right)\right) \\ &\leq (q_{2k-1} + q_{2k})\left(1 - \frac{q_{2k}}{2(q_{2k-1} + q_{2k})} - \frac{3}{4}\right) \\ &= \frac{1}{4}(q_{2k-1} - q_{2k}). \end{aligned} \quad (7)$$

□

Algorithm 1 Transformation of T_{Huffman} into T_{base}.

for $k = 2, \dots, K-1$ **do**
  if q_{2k-1} is a leaf and $2q_{2k} < q_{2k-1}$ **then**
    Convert the sibling pair (q_{2k}, q_{2k-1}) into a master node and its grandchild. $\cdots (*)$
  end if
end for

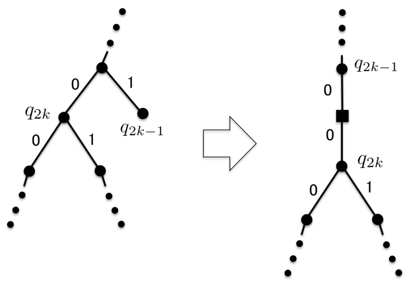


Fig. 4. The conversion from a sibling pair $(q_{2k-1},\ q_{2k})$ into a master node and its grandchild.

Now, consider a K-ary source and let T_{Huffman} be the corresponding Huffman code tree. We transform T_{Huffman} into a new tree T_{base} using Algorithm 1. The conversion shown by $(*)$ in Algorithm 1 is illustrated in Fig. 4. It is an operation that lifts up q_{2k-1} to make a master node and pulls down the entire subtree of q_{2k} to one lower level. Two nodes, q_{2k-1} and q_{2k}, are then connected by code symbols '00'. Let $\mathcal{K}$ be the set of indices whose corresponding sibling pairs are converted by Algorithm 1 and let $\mathcal{U}$ denote the set of indices of entire sibling pairs, so that

$$\mathcal{K} = \{k \in \{2,\dots,K-1\} | q_{2k-1} \text{ is a leaf and } 2q_{2k} < q_{2k-1}\},$$
$$\mathcal{U} = \{1,\dots,K-1\}.$$

Lemma 4. *For $\frac{1}{2} \le q_1 \le \frac{2}{3}$, the redundancy of optimal binary AIFV codes is upper bounded by $q_1^2 - 2q_1 + 2 - h(q_1)$.*

Proof. Let T_0 be T_{base} and transform T_{base} into T_1 by the operation described in Fig. 5. It is easy to see that T_0 and T_1 are valid binary AIFV code trees, satisfying all the properties mentioned in Section II-A.

The total probability assigned to master nodes is $\sum_{k\in\mathcal{K}} q_{2k-1}$ for both T_0 and T_1. Thus, it follows from the encoding procedure 2) in Section II-A that the transition probabilities, $P(T_1|T_0)$ and $P(T_0|T_1)$, are given by $\sum_{k\in\mathcal{K}} q_{2k-1}$ and $1 - \sum_{k\in\mathcal{K}} q_{2k-1}$, respectively. Therefore, the stationary probabilities, $P(T_0)$ and $P(T_1)$, are calculated as $1 - \sum_{k\in\mathcal{K}} q_{2k-1}$ and $\sum_{k\in\mathcal{K}} q_{2k-1}$, respectively. Then, we have from (1) and $L_{\text{OPT}} \le L_{\text{AIFV}}$ that

$$\begin{aligned} L_{\text{OPT}} &\le L_{T_0} \cdot P(T_0) + L_{T_1} \cdot P(T_1) \\ &= \left(\sum_{k\in\mathcal{U}\setminus\mathcal{K}} (q_{2k-1} + q_{2k}) + \sum_{k\in\mathcal{K}} 2q_{2k} \right) \cdot \left(1 - \sum_{k\in\mathcal{K}} q_{2k-1} \right) \\ &\quad + \left(q_2 + \sum_{k\in\mathcal{U}\setminus\mathcal{K}} (q_{2k-1} + q_{2k}) + \sum_{k\in\mathcal{K}} 2q_{2k} \right) \cdot \sum_{k\in\mathcal{K}} q_{2k-1}. \end{aligned} \tag{8}$$

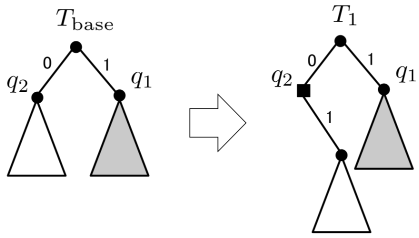


Fig. 5. Transformation of T_{base} into T_1.

Applying chain rules of entropy on T_{Huffman} from the root to leaves gives the following decomposition of the source entropy [5].

$$H(X) = \sum_{k=1}^{K-1} (q_{2k-1} + q_{2k})\, h\left(\frac{q_{2k}}{q_{2k-1} + q_{2k}} \right). \tag{9}$$

Thus, the redundancy of the optimal binary AIFV code, r_{AIFV}, defined by (2) is upper bounded from (8) and (9) as follows.

$$\begin{aligned} r_{\text{AIFV}} &\le L_{T_0} \cdot P(T_0) + L_{T_1} \cdot P(T_1) - H(X) \\ &= \left[q_1 + q_2 - h\left(\frac{q_2}{q_1+q_2} \right) \right] \\ &\quad + \sum_{k\in\mathcal{K}} \left[2q_{2k} - (q_{2k} + q_{2k-1}) h\left(\frac{q_{2k}}{q_{2k-1}+q_{2k}} \right) + q_2 q_{2k-1} \right] \\ &\quad + \sum_{k\in\mathcal{U}\setminus(\mathcal{K}\cup\{1\})} (q_{2k-1} + q_{2k}) \left(1 - h\left(\frac{q_{2k}}{q_{2k-1}+q_{2k}} \right) \right). \end{aligned} \tag{10}$$

Note that in (10), we decompose the sum on $\mathcal{U}$ into three terms each of which is summed over $\{1\}$, $\mathcal{K}$, and $\mathcal{U}\setminus(\mathcal{K}\cup\{1\})$. First, suppose $k \in \mathcal{K}$. It follows from the definition of $\mathcal{K}$ that $2q_{2k} < q_{2k-1}$. Thus, we can apply Lemma 2 with $w_1 := q_{2k-1}$, $w_2 := q_{2k}$ and $q := q_2 \le \frac{1}{2}$ to each $k \in \mathcal{K}$. Next, suppose $k \in \mathcal{U} \setminus (\mathcal{K} \cup \{1\})$. If q_{2k-1} is a leaf, $q_{2k-1} \le 2q_{2k}$ holds since $k \notin \mathcal{K}$. If q_{2k-1} is not a leaf, then by Lemma 1, $q_{2k-1} \le 2q_{2k}$ holds. In either case, $q_{2k-1} \le 2q_{2k}$ holds. Thus, we can apply Lemma 3 to each $k \in \mathcal{U}\setminus(\mathcal{K}\cup\{1\})$. Combining these with $q_1 + q_2 = 1$, we get

$$\begin{aligned} r_{\text{AIFV}} &\le 1 - h(q_1) + \sum_{k\in\mathcal{K}} q_2 (q_{2k-1} - q_{2k}) \\ &\quad + \sum_{k\in\mathcal{U}\setminus(\mathcal{K}\cup\{1\})} \frac{1}{4} (q_{2k-1} - q_{2k}) \\ &< 1 - h(q_1) + q_2 \sum_{k=2}^{K-1} (q_{2k-1} - q_{2k}) && (11) \\ &< 1 - h(q_1) + q_2 q_3 && (12) \\ &\le 1 - h(q_1) + q_2^2 && (13) \\ &= q_1^2 - 2q_1 + 2 - h(q_1). && (14) \end{aligned}$$

Ineq. (11) holds since $\frac{1}{4} < \frac{1}{3} \le 1 - q_1 = q_2$. Ineqs. (12) and (13) hold since the sequence $\{q_k\}$ is non-increasing. □

Proof of Theorem 3. First, consider the case of $\frac{1}{2} \le p_{\max} \le \frac{-1+\sqrt{5}}{2}$. Since $p_{\max} = q_1$ for $p_{\max} \ge \frac{1}{2}$, it follows that $\frac{1}{2} \le$

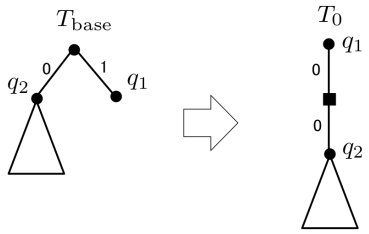


Fig. 6. Transformation of $T_{\rm base}$ into T_0.

$q_1 \leq \frac{-1+\sqrt{5}}{2} < \frac{2}{3}$. Applying Lemma 4, we get the upper bound on the redundancy as $p_{\max}^2 - 2p_{\max} + 2 - h(p_{\max})$.

Next, we prove the bound for $\frac{-1+\sqrt{5}}{2} \leq p_{\max} < 1$. The proof follows the same line as the proof of Lemma 4. First, transform $T_{\rm base}$ into T_0 by the operation depicted in Fig. 6 and also transform $T_{\rm base}$ into T_1 as illustrated in Fig. 5.

Then, T_0 and T_1 are valid binary AIFV code trees. In the same way as Lemma 4, we can show that the stationary probabilities are given by $P(T_0) = \frac{1-\sum_{k\in\mathcal{K}} q_{2k-1}}{1+q_1}$ and $P(T_1) = \frac{q_1+\sum_{k\in\mathcal{K}} q_{2k-1}}{1+q_1}$. As before, the redundancy of the optimal binary AIFV code can be upper bounded as follows.

$$\begin{aligned} r_{\rm AIFV} &\leq L_{T_0}\cdot P(T_0) + L_{T_1}\cdot P(T_1) - H(X) \\ &= \left[2q_2 + \frac{q_1^2}{1+q_1} - h\left(\frac{q_2}{q_1+q_2}\right)\right] \\ &\quad + \sum_{k\in\mathcal{K}}\left[2q_{2k} - (q_{2k}+q_{2k-1})h\left(\frac{q_{2k}}{q_{2k-1}+q_{2k}}\right) + \frac{q_1 q_{2k-1}}{1+q_1}\right] \\ &\quad + \sum_{k\in\mathcal{U}\setminus(\mathcal{K}\cup\{1\})}(q_{2k-1}+q_{2k})\left(1 - h\left(\frac{q_{2k}}{q_{2k-1}+q_{2k}}\right)\right). \end{aligned} \tag{15}$$

We can apply Lemma 2 with $w_1 := q_{2k-1},\ w_2 := q_{2k}$ and $q := \frac{q_1}{1+q_1} \leq \frac{1}{2}$ to the second term of (15). Also, we can apply Lemma 3 to the third term of (15). Combining with $q_1+q_2 = 1$ and $q_1 = p_{\max}$, we get

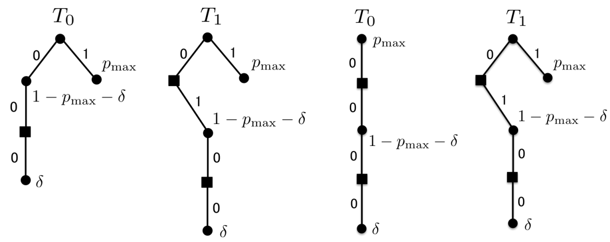
$$\begin{aligned} r_{\rm AIFV} &\leq \left[2(1-q_1) + \frac{q_1^2}{1+q_1} - h(q_1)\right] \\ &\quad + \sum_{k\in\mathcal{K}}\frac{q_1}{1+q_1}(q_{2k-1}-q_{2k}) + \sum_{k\in\mathcal{U}\setminus(\mathcal{K}\cup\{1\})}\frac{1}{4}(q_{2k-1}-q_{2k}) \\ &\leq 2(1-q_1) + \frac{q_1^2}{1+q_1} - h(q_1) + \frac{q_1}{1+q_1}\sum_{k=2}^{K-1}(q_{2k-1}-q_{2k}) \end{aligned} \tag{16}$$

$$\leq 2(1-q_1) + \frac{q_1^2}{1+q_1} - h(q_1) + \frac{q_1 q_2}{1+q_1} \tag{17}$$

$$= \frac{-2p_{\max}^2 + p_{\max} + 2}{1+p_{\max}} - h(p_{\max}). \tag{18}$$

Ineq. (16) holds since $\frac{1}{4} < \frac{3-\sqrt{5}}{2} \leq \frac{q_1}{1+q_1}$. Ineq. (17) holds since the sequence $\{q_k\}$ is non-increasing.

To prove that the derived bound is tight, it is sufficient to show that there exists a source for every $p_{\max} \geq \frac{1}{2}$ such that the source attains the bound arbitrarily closely. In particular, we show that a source with probabilities $(p_{\max}, 1-p_{\max}-\delta, \delta)$ satisfies the bound with equality in the limit of $\delta \to 0$. Note that for $|\mathcal{X}| = 3$, there exist only four possible tree structures for each code tree, T_0 and T_1. By examining all the possible combinations of the structures, it can be shown that the optimal binary AIFV codes are as illustrated in Fig. 7 for each range of $p_{\max}$. We see that the redundancy of the codes coincides with the bound in the limit of $\delta \to 0$. □



(a) $\frac{1}{2} \leq p_{\max} \leq \frac{\sqrt{5}-1}{2}$. (b) $\frac{\sqrt{5}-1}{2} \leq p_{\max} \leq 1$.

Fig. 7. The bound achieving trees.

Proof of Theorem 4. In the case of $p_{\max} < \frac{1}{2}$, we note that $|\mathcal{X}| \geq 3$ and thus, q_1 must be an internal node. It follows from Lemma 1 that $q_1 \leq 2q_2$. Since $q_1 + q_2 = 1$, we get $\frac{1}{2} \leq q_1 \leq \frac{2}{3}$. By Lemma 4, we obtain

$$r_{\rm AIFV} < \max_{\frac{1}{2}\leq q_1\leq\frac{2}{3}} q_1^2 - 2q_1 + 2 - h(q_1) = \frac{1}{4}. \tag{19}$$

□

V. Conclusion

In this paper, we considered binary AIFV codes that use two code trees and decoding delay is at most two bits. We showed that the redundancy of the codes is at most $\frac{1}{2}$. Furthermore, we can expect that if the codes are allowed to use more trees and the decoding delay can be more than two bits, the upper bound on the redundancy can further be improved. It is also interesting to derive tighter upper bounds on the redundancy of optimal binary AIFV codes for $p_{\max} < \frac{1}{2}$, and compare them to their Huffman counterparts.

References

[1] B. McMillan, "Two inequalities implied by unique decipherability," *Inform. Theory, IRE Trans.*, vol. IT-2, no. 4, pp. 115–116, 1956.

[2] D. A. Huffman, "A method for the construction of minimum redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[3] H. Yamamoto and X. Wei, "Almost instantaneous FV codes," in *Proceedings of the 2013 IEEE ISIT, Istanbul, Turkey*, 2013, pp. 1759–1763.

[4] H. Yamamoto, M. Tsuchihashi, and J. Honda, "Almost instantaneous fixed-to-variable length codes," *IEEE Trans. Inform. Theory*, vol. 61, no. 12, pp. 6432–6443, 2015.

[5] R. G. Gallager, "Variations on a theme by Huffman," *IEEE Trans. Inform. Theory*, vol. IT-24, no. 6, pp. 668–674, 1978.

[6] B. L. Montgomery and J. Abrahams, "On the redundancy of optimal binary prefix-condition codes for finite and infinite sources," *IEEE Trans. Inform. Theory*, vol. 33, no. 1, pp. 156–160, 1987.

[7] D. Manstetten, "Tight bounds on the redundancy of Huffman codes," *IEEE Trans. Inform. Theory*, vol. 38, no. 1, pp. 144–151, 1992.

[8] R. M. Capocelli and A. D. Santis, "Tight upper bounds on the redundancy of Huffman codes," *IEEE Trans. Inform. Theory*, vol. 35, no. 5, pp. 1084–1091, 1989.

[9] O. Johnsen, "On the redundancy of binary Huffman codes," *IEEE Trans. Inform. Theory*, vol. IT-26, no. 2, pp. 220–222, 1980.

[10] C. Ye and R. W. Yeung, "A simple upper bound on the redundancy of Huffman codes," *IEEE Trans. Inform. Theory*, vol. 48, no. 7, pp. 2132–2138, 2002.