Two Level Hamming-Huffman Trees

Min C. Lin [1] Fabiano S. Oliveira [2] Paulo E. D. Pinto [2] Moysés S. Sampaio Jr [3] Jayme L. Szwarcfiter [2,3]

[1]Universidad de Buenos Aires, Argentina [2]Universidade do Estado do Rio de Janeiro, Brazil [3]Universidade Federal do Rio de Janeiro, Brazil

oscarlin@dc.uba.ar {fabiano.oliveira,pauloedp}@ime.uerj.br {moysessj,jayme}@cos.ufrj.br

remote 9th LAWCG and MDA

November 25th, 2020

Introduction

In information theory, there is a common trade-off that arises in data transmission processes, in which two goals are usually tackled independently: data compression and preparation for error detection. While data compression shrinks the message as much as possible, data preparation for error detection adds redundancy to messages so that a receiver can detect, or fix, corrupted ones. Data compression can be achieved using different strategies, often depending on the type of data being compressed. One of the most traditional methods is the method of Huffman [1], that uses ordered trees, known as Huffman trees, to encode the symbols of a given message. In 1980, Hamming proposed the union of both compression and error detection through a data structure called Hamming-Huffman tree [2], which extends the Huffman tree by allowing the detection of any 1-bit transmission error. Determining optimal Hamming-Huffman trees is still an open problem.

Contribution

In this work, we describe an algorithm to determine optimal two level Hamming-Huffman trees when the symbols have uniform frequencies. That is, the algorithm builds optimal Hamming-Huffman trees in which all leaves lay in at most two different levels. Also, considering experimental results, we conjecture that, for uniform frequencies, optimal two levels Hamming-Huffman trees are optimal in general.

Hamming-Huffman Trees

A *Huffman tree* (HT) T is a rooted strict binary tree in which each edge (u, v), v being a left (resp. right) child of u, is labeled by 0 (resp. 1) and there is a one-to-one mapping between the set of leaves of T and the set Σ of symbols of the message M to be sent. Given T, each symbol a of M is encoded into a binary string $c(a)$. Such encoding is obtained by the directed path from the root of T to the leaf corresponding to a. Over all possible trees, the HT for M is a tree in which its cost, defined as the sum of $p(a)|c(a)|$ over all $a \in \Sigma$, is minimized, where $p(a)$ stands for the probability of occurrence of a and $|c(a)|$ is the length of the string $c(a)$.

A *Hamming-Huffman tree* (HHT) T is an extension of the HT in which, for each leaf labeled with $a \in \Sigma$, there exist leaves $e_1, \ldots, e_k$ with $k = |c(a)|$ such that each $c(e_i)$, $1 \leq i \leq k$, differs from $c(a)$ in exactly one position. The leaves $e_1, \ldots, e_k$ are called *error leaves* of a. When $c(e)$ is identified during the decoding process, where e is an error leaf, it means that a transmission error is detected. The cost of HHT's is defined exactly in the same way as the cost of HTs. We define an HHT as *optimal* if its cost is minimum.

Figure 1 depicts an HT with cost 2.4 and an optimal HHT with cost 3.8, both having 5 symbols with uniform frequencies, that is, symbols with a same probability of occurrence.

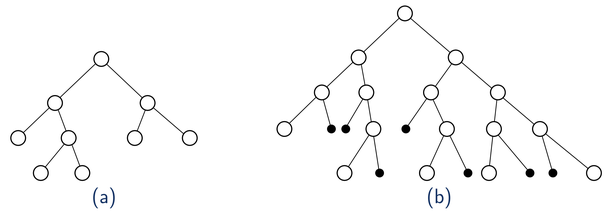

Figure 1: Examples of (a) Huffman and (b) optimal Hamming-Huffman trees, for 5 symbols with uniform frequencies. White (resp. black) leaves represent symbol (resp. error) leaves.

Hamming-Huffman trees with leaves in two levels

Consider the problem of finding an optimal HHT for ℓ uniform-frequency symbols such that these symbols are placed on at most two levels. We will describe an efficient algorithm for this problem. There is a one-to-one mapping between the leaves of a full binary HHT having height n and the vertices of an hypercube Q_n, in which a leaf a corresponds to $c(a) \in V(Q_n)$. The problem of finding the minimum number of error leaves in a full binary HHT T, of height n with ℓ symbol leaves, is equivalent to that of finding one that minimizes $|N(L)|$, over all independent sets L of cardinality ℓ in n-cubes. Define $\varphi(\ell, n)$ as this minimum value.

Concerning an optimal HHT with leaves on two levels $h_1 < h_2$, consider that there are ℓ_1 symbol leaves on level h_1, for some $1 \leq h_1 \leq \lceil \log \ell \rceil + 1$ and $1 \leq \ell_1 \leq \min\{\ell, 2^{h_1-1}\}$. Therefore, the minimum number of error leaves is $\varphi(\ell_1, h_1)$ and thus $r(\ell_1, h_1) = 2^{h_1} - (\ell_1 + \varphi(\ell_1, h_1))$ is the number of leaves that neither are symbol nor error leaves. The remaining $\ell_2 = \ell - \ell_1$ symbols are distributed among the subtrees rooted at these $r(\ell_1, h_1)$ leaves. To accomplish this, each subtree is required to have precisely height $h'(\ell_1, h_1) = \lceil \log \frac{\ell_2}{r(\ell_1,h_1)} \rceil + 1$. The strategy is to choose among all the possible trees, one that has minimum cost. Given h_1, ℓ_1 and ℓ, the cost of each tree is given by

$$T(h_1, \ell_1, \ell) = \begin{cases} \ell h_1, \text{ if } \ell = \ell_1 \\ +\infty, \text{ if } r(\ell_1, h_1) = 0 \text{ and } \ell > \ell_1 \\ \ell h_1 + \ell_2 h'(\ell_1, h_1), \text{ otherwise.} \end{cases}$$

The cost of an optimal tree for ℓ symbols can be obtained by

$$\min\{T(h_1, \ell_1, \ell) : 1 \leq h_1 \leq \lceil \log \ell \rceil + 1 \ , \ 1 \leq \ell_1 \leq \min\{\ell, 2^{h_1-1}\}\}$$

Concerning the complexity, for each h_1, there are at most 2^{h_1-1} possible values for ℓ_1. Therefore, there are at most $1 + 2 + 2^2 + \ldots + 2^{\lceil \log \ell \rceil} = \Theta(\ell)$ distinct pairs of values h_1, ℓ_1 to be computed for T. Moreover, for each computation of $T(h_1, \ell_1, \ell)$, the evaluation of $\varphi(\ell_1, h_1)$ is required, which can be done in time $O(h_1^2)$ [3]. So, the complexity of the method is $O(\ell \log^2 \ell)$. Figure 2 depicts this strategy. Nodes with "s" represent symbol leaves, black nodes represent the error leaves, and dashed nodes represent the free leaves.

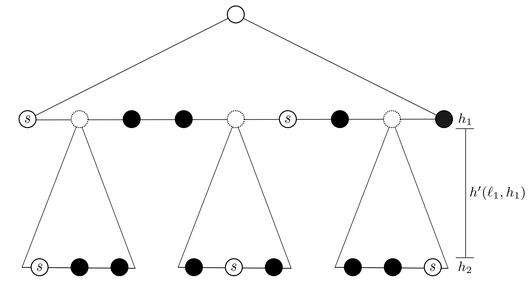

Figure 2: Hamming-Huffman tree with leaves on two levels h_1 and h_2.

Regarding general Hamming-Huffman trees

We have implemented two algorithms. The first one is a backtracking that finds an optimal Hamming-Huffman tree. The second one is a dynamic programming algorithm that evaluates a lower bound for the cost of an optimal Hamming-Huffman tree. Both consider ℓ symbols with uniform frequencies. With respect to the backtracking, we have tested all values of $1 \leq \ell \leq 38$, concluding that there is always an optimal Hamming-Huffman tree with at most two levels. Concerning the dynamic programming algorithm, we have tested all values of $1 \leq \ell \leq 400$. We have verified that, for some cases, the lower bound was equal to the cost of the corresponding optimal two level HHT's.

Considering the results of the experiments, we believe that optimal HHT's for symbols with uniform frequencies indeed have leaves on at most two levels, as formalized in the following conjecture.

Conjecture

Let Σ be a set of symbols having a same frequency. There exists an optimal Hamming-Huffman tree associated with Σ in which all leaves are on at most two levels.

References

[1] David A. Huffman.
A method for the construction of minimum redundancy codes.
Proceedings of the IRE, 40:1098–1101, 1951.

[2] Richard W. Hamming.
Coding and Information Theory.
Prentice-Hall, 1986.

[3] Janos Körner and Victor K. Wei.
Odd and even Hamming spheres also have minimum boundary.
Discrete Mathematics, 51:147–165, 1984.