The meet operation in the imbalance lattice of maximal instantaneous codes: alternative proof of existence

Stephan Foldes, D. Stott Parker, Sándor Radeleczki

Abstract

An alternative proof is given of the existence of greatest lower bounds in the imbalance order of binary maximal instantaneous codes of a given size. These codes are viewed as maximal antichains of a given size in the infinite binary tree of 0-1 words. The proof proposed makes use of a single balancing operation instead of expansion and contraction as in the original proof of the existence of glb.

1 Terminology of codes and introduction

The set $\{0,1\}^*$ of all finite sequences (words) of the symbols 0 and 1 is partially ordered by the *prefix order* $\leq_{pref}$ defined by

$$v \leq_{pref} w \Leftrightarrow \exists z \ vz = w$$

The prefix-ordered set of words is an infinite binary tree having the empty word as root. *Instantaneous code*s are defined as the finite antichains in this tree. (This finiteness shall be assumed throughout the paper, thus excluding infinite prefix-free sets.) By *lexicographic (lex) order* we mean the (only) linear extension of the prefix order in which words incomparable in the prefix order are compared by the *"telephone book principle"*, i.e. $v0x$ always precedes (is smaller than) $v1y$. We call an instantaneous code *lex monotone* if the sequence of lengths of the codewords taken in lex order is monotone (non-decreasing). With respect to a given lex monotone instantaneous code C whose codewords in lex order are $c_1, c_2, ...$, each codeword c_i is then identified by its *index* i.

[0]Keywords and phrases: Instantaneous code, prefix code, path-length sequence, imbalance lattice, balancing operation, Kraft sum, binary tree, canonical tree

[0]AMS Classification 2010: Primary: 06A07, 94A45

It is well known that for every maximal instantaneous code there is a unique lex monotone maximal instantaneous code with the same multiset of codeword lengths. (See e.g. $[FS]$ for more general statements.) Thus the multiset of codeword lengths, displayed as a list of numbers with repetitions in non-decreasing order, can be used to denote the maximal instantaneous lex monotone code, these are the *path-length sequences* appearing in $[SPPR]$. For example, the only lex monotone maximal instantaneous code of size 3 is $\{0, 10, 11\}$, and its path-length sequence is $(1, 2, 2)$. The code can also be displayed by the binary tree of all prefixes of the codewords (called a *canonical tree* by Elsholtz, Heuberger and Prodinger in $[EHP]$, under assumption of lex monotonicity), and the path-length sequence is then the sequence of lengths of root-to-leaf paths of this tree.

For any word w we can use the simplified notation

$$2^{-w}$$

to denote the number obtained by raising $1/2$ to a power equal to the length of w. The *Kraft sum* of any instantaneous code C is then the sum

$$K(C) = \sum_{w \in C} 2^{-w}$$

The Kraft sum is always at most 1, and it is equal to 1 if and only if the instantaneous code C is maximal (Kraft $[K]$).

2 Statements and proofs

With the above terminology and notation we rephrase the definition of imbalance order and the result that it is a lattice (Stott Parker and Prasad Ram $[SPPR]$) as follows.

Definition of Imbalance Order by Majorization $[SPPR]$ *Let L be the set of lex monotone maximal instantaneous codes of a same given size. For codes $A, B \in L$, lexicographically enumerated*

$$A = a_1, a_2, ... \qquad B = b_1, b_2, ... \tag{1}$$

A is said to be more balanced (less imbalanced) than or equal to B, in symbols $A \trianglelefteq B$, if for all $m \geq 1$ we have the following inequality for the partial Kraft sums:

$$K(a_1, ..., a_m) \leq K(b_1, ..., b_m)$$

A characterization of the imbalance order via ternary exchanges was also given in $[SPPR]$, and in the sequel we shall give another characterization by comparing indices in the enumerations (1).

The size $f(t)$ of the poset of lex monotone maximal instantaneous codes of size t is an exponentially growing function of the parameter t, which has been the object of combinatorial studies since the 1960's (see $[EHP]$, where further references are also given). While a closed formula for $f(t)$ is not available, Elsholtz, Heuberger and Prodinger $[EHP]$ gave a new and very tight asymptotic estimate of $f(t)$.

Lattice Property of Imbalance Order $[SPPR]$ *The imbalance-ordered set of lex monotone maximal instantaneous codes of the same given size is a lattice.*

Due to the lattice property the construction of optimal codes becomes an optimization problem on a lattice. In a context very different from that of binary codes, the balance concept introduced in $[SPPR]$ has also been shown by O'Keefe, Pajoohesh and Schellekens $[OKPS]$ to be relevant in studying the efficiency of algorithms that involve a bifurcation at each step, as a root-to-leaf path in the decision tree of an algorithm corresponds to the succession of steps of the algorithm on a particular input, and path-length corresponds to running time on that input. Besides pointing to the analogy between the concepts of average codeword length and average running time, $[OKPS]$ also shows that, with the exception of the very small lattices, the imbalance lattices are not modular. Pajoohesh $[P]$ characterizes the most balanced and the most imbalanced trees in terms of the semilattice structure of the trees themselves.

In the present paper an alternative proof of the above lattice property is given, based not on induction on the common size of the codes in the imbalance-ordered set of codes, but on applying the abstract Criterion below for a poset to be a lattice. An earlier alternative explanation of the lattice property, in fact closer to the techniques of the original proof of the result in $[SPPR]$, was given by two of the present authors in $[FR]$.

Criterion for Lattice Property *For any finite partially ordered set with minimum and maximum the following conditions are equivalent:*

(i) the poset is a lattice,

(ii) for every pair of distinct elements b, c, one of them - say c - can be replaced by a lesser element $d < c$, such that b and c have the same common lower bounds as b and d,

(iii) for every pair of distinct elements b, c there is an element d that is less than b or c, and such that b, c, d have the same common lower bounds as b, c.

Proof Condition (ii) obviously holds in any lattice, while the existence of a greatest lower bound of b and c is obtained, using condition (ii), by induction on the number of elements that are below at least one of b and c. Condition (iii) is a re-phrasing of (ii). $\square$

The characterization of the imbalance order given below, by comparing indices, and a reduction lemma, shall make the above Criterion applicable. The characterization is based on the following description of the comparabilities between elements of any two finite maximal instantaneous codes:

Interval Decomposition Lemma for Two Codes *For any two maximal instantaneous codes A and B there is a unique positive integer n and unique partitions of the lexicographically ordered codes into n pairwise disjoint non-empty intervals consecutive in the lexicographic order*

$$\begin{array}{ll} A = A_1 \cup ... \cup A_n & B = B_1 \cup ... \cup B_n \\ A_1 < ... < A_n & B_1 < ... < B_n \end{array} \tag{2}$$

such that any words $x \in A_i$ and $y \in B_j$ are comparable in the prefix order if and only if $i = j$.

Proof Two elements of A belong to the same interval if and only if there is some element of B comparable with both. The intervals of B are defined similarly. The fact that this defines interval decompositions of the two codes with the same number n of intervals is verified without difficulty. The claimed properties and uniqueness are also straightforward. $\square$

The interval decompositions (2) also have the following properties:

(i) for every i, at least one of A_i or B_i is a singleton, both are singletons if and only if $A_i = B_i$, otherwise they are disjoint,

(ii) if the interval A_i (respectively B_i) is a singleton, then its unique element is a prefix of the words in B_i (respectively in A_i),

(iii) for every i we have the equality of the corresponding interval Kraft sums, $K(A_i) = K(B_i)$,

(iv) if $i < j$, $x \in A_i \cup B_i$, $y \in A_j \cup B_j$, then x and y are incomparable in the prefix order and x precedes y lexicographically.

With a view of referring to these interval decompositions in the sequel, we call the intervals A_i (respectively B_i) in (2) the *(comparability) blocks* of A with respect to B (of B with respect to A). A block A_i (respectively B_i) is said to be *dominating* if it is a singleton but B_i (respectively A_i) is not. In that case the sole element of A_i (respectively of B_i) is a proper prefix of every word in B_i (respectively in A_i). Note that if A_i and B_i are not coinciding singletons, then exactly one of them is a dominating block.

Characterization of the Imbalance Order by Comparing Indices *For lexicographically enumerated maximal instantaneous codes*

$$A = a_1, a_2, ... \qquad B = b_1, b_2, ...$$

of the same size, we have $A \trianglelefteq B$ in the imbalance order (A is more balanced than or equal to B) if and only if whenever a_i and b_j are comparable codewords in the prefix order, for their indices we have $i \geq j$.

Proof Suppose that $A \trianglelefteq B$ and for some codewords a_i and b_j comparable in the prefix order we have $i < j$. We shall derive a contradiction. Consider the interval decompositions for the two codes, as in (2). Due to the comparability of the two codewords, they belong to corresponding intervals, *i.e.* there is an index k such that $a_i \in A_k$ and $b_j \in B_k$. If A_k is a singleton, $A_k = \{a_i\}$, then

$$K(a_1, ..., a_j) > K(a_1, ..., a_i) = K(A_1 \cup ... \cup A_k) = K(B_1 \cup ... \cup B_k) \geq K(b_1, ..., b_j)$$

contradicts majorization. If B_k is a singleton, $B_k = \{b_j\}$, then majorization is contradicted by

$$K(a_1, ..., a_i) > K(A_1 \cup ... \cup A_{k-1}) = K(B_1 \cup ... \cup B_{k-1}) = K(b_1, ..., b_{j-1}) \geq K(b_1, ..., b_i)$$

Suppose conversely that whenever a_i and b_j are comparable codewords in the prefix order, for their indices we have $i \geq j$, but majorization fails for some index m,

$$K(a_1, ..., a_m) > K(b_1, ..., b_m) \tag{3}$$

Let the indices k, l be determined by $a_m \in A_k, b_m \in B_l$. If A_k is a singleton, $A_k = \{a_m\}$, then (3) requires that the lex last word b_u in B_k lexicographically follow b_m. Then $m < u$ contradicts the comparability of a_m and b_u. If B_l is a singleton, $B_l = \{b_m\}$, then (3) requires that a_m lexicographically follow all words in A_l. Let a_i be any word in A_l. Now $i < m$ contradicts the comparability of a_i and b_m. $\square$

Reduction Lemma *If B, C are two distinct lexicographically monotone maximal instantaneous codes of the same size, then there is a lex monotone maximal instantaneous code D that is (strictly) more balanced than at least one of B or C, and such that in the imbalance order B, C, D have the same common lower bounds as B, C.*

Proof Let the given codes be enumerated in lex order as

$$B = b_1, b_2, ... \qquad C = c_1, c_2, ... \tag{4}$$

Since B and C are distinct, there must exist elements $b \in B$ and $c \in C$ such that

(i) c is a proper prefix of some element of B

(ii) b is a proper prefix of some element of C.

Without loss of generality we can assume that the first such c lexicographically precedes the first such b. Let k denote the index in B of the lexicographically first element b satisfying condition (ii). With k thus fixed, let m denote the index in C of the lexicographically last element c among those elements of C which lexicographically precede b_k and satisfy condition *(i)*. Thus a word c_m in C has also been chosen, and it is easy to see that $m < k$.

With reference to the terminology of decompositions according to the Interval Decomposition Lemma for Two Codes, b_k is the sole element of the first dominating block of B with respect to C, and c_m is the sole element of the last block of C that is dominating and precedes all non-singleton blocks of C.

The code D is now constructed as follows. It is obtained from C by a single balancing operation, in the sense of $[SPPR]$, chosen to take into account the relationship of C with B, and the choice of the codewords c_m and b_k. Referring to the indexed enumeration of C in lex order appearing in (4), let c_n, c_{n+1} be the first two among the elements of C admitting b_k as a prefix which have the same length. It is not difficult to see that $m < n$ and c_n, c_{n+1} are twin sons (one letter extensions) of some word w, $c_n = w0, c_{n+1} = w1$.

Let $D = (C \setminus \{c_m, c_n, c_{n+1}\}) \cup \{w, c_m 0, c_m 1\}$. Obviously in the imbalance order $D \trianglelefteq C$.

We claim that if an arbitrary lex monotone maximal instantaneous code A with lex enumerated codewords $a_1, a_2, ...$ is more balanced than (or equal to) B and C, then it is also more balanced than or equal to D. This will show that the statement of the Lemma holds.

The elements of D, enumerated as a sequence of words in lex order as $d_1, d_2, ...$, are partitioned into five consecutive subsequences:

$d_1 = c_1$, ..., $d_{m-1} = c_{m-1}$ (empty subsequence if $m = 1$)

$d_m = c_m 0$, $d_{m+1} = c_m 1$

$d_{m+2} = c_{m+1}$,......, $d_n = c_{n-1}$

(empty subsequence if $m+1 = n$, non-empty if $m+1 < n$)

$d_{n+1} = w$

$d_{n+2} = c_{n+2}$, $d_{n+3} = c_{n+3}$,....

(empty subsequence if $n+1$ is the common size of the codes)

The subsequence $d_{m+2}, ..., d_n$ in turn consists of two (possibly empty) consecutive subsequences: the first of these consists of elements also belonging to B, and the second consists of elements that have b_k as a proper prefix. In the first subsequence the index in B of any element is (strictly) larger than its index in C. In the second subsequence the last symbol of each element is 0 (i.e. it is not of the form $d_h = v1$, for otherwise $v0$ would have to be also in this second subsequence, contradicting the definition of n).

In view of the Characterization of the Imbalance Order by Comparing Indices, we need to verify that if some codeword a in A is comparable in the prefix order to a codeword in D, i.e. to some d_j having index j in the lex enumeration of D, then the index of a in A is at least j. This is obvious for $j < m$ and $n + 1 < j$, since $A \trianglelefteq B, C$. In the following examination of the remaining cases *comparability* will always refer to comparability in the prefix order of the tree of words.

For $j = m$, if an element a_i of A is comparable to $d_m = c_m 0$, then it is also comparable to its prefix c_m. The assumption $A \trianglelefteq C$ then implies $i \geq m$.

For $j = m + 1$, if an element a of A is comparable to $d_{m+1} = c_m 1$, it is of course also comparable to c_m, and we claim that it is comparable as well to some member b_h of B with index $h > m$. Note that both $d_m = c_m0$ and $d_{m+1} = c_m1$ must be the prefixes of words in B, and no two of the codewords in C can be prefixes of the same word in B (while each one of $c_1, ..., c_{m-1}$ is the prefix of at least one word in B and c_m is a prefix of at least two). From this we can conclude that d_{m+1} must be the prefix of some b_h in B, and all such elements of B have index $h > m$. Now it follows that the element a of A is comparable to at least one such b_h with index $h > m$. But then, as $A \trianglelefteq B$ in the imbalance order, the index of a in A is at least $h \geqslant m + 1$.

In the interval $m + 2 \leq j \leq n$, if it is not empty, let j be the smallest index such that for some $i < j$ the elements a_i and d_j are comparable - we shall derive a contradiction. For j thus fixed, let i be as small as possible. If $d_j = c_{j-1}$ belongs to B, then its index in B is (strictly) greater than $j - 1$, thus by $A \trianglelefteq B$ in the imbalance order it could not be comparable to a_i. Therefore $d_j = c_{j-1}$ has b_k as a proper prefix. Also its length is (strictly) larger than that of c_{j-2}. This implies that the last symbol of d_j must be 0. But now, since the last symbol of d_j is 0, if the word a_i were a proper prefix of $d_j = c_{j-1}$, then it would be a proper prefix of c_j also, implying $i \geq j$, which is contrary to assumption. Thus d_j is a prefix of a_i and d_j is the sole element of D comparable to a_i. Now d_{j-1} is comparable to one or more elements a_r of A, and all such indices r must be at least $j-1$ by the minimality assumption on j. But as a_i cannot be comparable with d_{j-1}, it must come later in the lex order on A than all the elements a_r of A comparable with d_{j-1}, i.e. $r < i$ for all such r. Therefore $i > r \geq j - 1$ and thus i is at least j, a contradiction.

The argument is similar for $j = n+1$. If the element a_i of A comparable with $d_{n+1} = w$ is comparable with $c_{n+1} = w1$ then we are done. Else a_i must have $c_n = w0$ and $w = d_{n+1}$ as a prefix and cannot be comparable with d_n. Therefore, $i > r \geq j - 1$ and thus a_i must come later in the lex order on A than all the elements of A comparable with d_n, i.e. $r < i$ for all such r. But we already know that the indices in A of these latter elements a_r are at least n, forcing $i \geqslant n + 1$.

We have thus shown that in the imbalance order all common lower bounds A of B and C are also lower bounds of the code D constructed from these latter two, completing the proof of the Lemma and thus providing an alternative proof of the Lattice Property. $\square$

Remark. Repeated application of the construction of D in the proof of the Reduction Lemma provides an algorithm for constructing the meet of any two codes B and C in the imbalance lattice. (The repetition is to be applied to the reduced pair of codes gotten by replacing B or C by D, according to whether D is more balanced than B or C.) As simple examples with incomparable B and C, we can take, using the lattice diagrams on p. 7. of $[SPPR]$, with path-length sequence representation of codes of size 7,

$B = 2\ 2\ 2\ 3\ 4\ 5\ 5 \quad C = 1\ 3\ 3\ 4\ 4\ 4\ 4 \quad D = 2\ 2\ 2\ 4\ 4\ 4\ 4$
(where D is in fact the meet of B and C)

or we can take as example of codes of size 9

$B = 233\ 333\ 455 \qquad C = 144\ 444\ 444 \qquad D = 223\ 444\ 444$
(where D is still less balanced than the meet of B and C).

Acknowledgements.
Part of this work has been co-funded by Marie Curie Actions and supported by the National Development Agency (NDA) of Hungary and the Hungarian Scientific Research Fund (OTKA, contract number 84593), within a project hosted by the University of Miskolc, Department of Analysis.

References

$[EHP]$ C. Elsholtz, C. Heuberger, H. Prodinger, The number of Huffman codes, compact trees, and sums of unit fractions, *IEEE Trans. Information Theory,* 59 (2) 2013, 1065-1075

$[FR]$ S. Foldes, S. Radeleczki, On the imbalance lattice of path-length sequences of binary trees, ArXiv 2013 (https://arxiv.org/abs/1307.0161)

$[FS]$ S. Foldes, N.M. Singhi, On instantaneous codes, Journal of Combinatorics, Information & System Sciences 31 (2006) 317–326

[K] L.G. Kraft, *A Device for Quantizing, Grouping, and Coding Amplitude Modulated Pulses*, Q.S. Thesis, MIT 1949

[$OKPS$] M. O'Keefe, H. Pajoohesh, M. Schellekens, Decision trees of algorithms and a semivaluation to measure their distance, *Electr. Notes Comput. Sc.* 161 (2006) 175-183

[P] H. Pajoohesh, Topological and categirical properties of of binary trees, *Applied Gen. Topology* 9 (1) (2008) 1-14

[$SPPR$] D. Stott Parker, Prasad Ram, The Construction of Huffman Codes is a Submodular ("Convex") Optimization Problem Over a Lattice of Binary Trees. *SIAM J. Comput.* 28(5) 1875-1905 (1999)

Authors' addresses:

S. Foldes
`stephan@renyi.mta.hu`

D.S. Parker
UCLA Computer Science Department
`stott@cs.ucla.edu`

S. Radeleczki
University of Miskolc, Institute of Mathematics
`matradi@uni-miskolc.hu`