Twenty (short) questions

Yuval Dagan[*], Yuval Filmus[†], Ariel Gabizon[‡], and Shay Moran[§]

July 18, 2018

Abstract

A basic combinatorial interpretation of Shannon's entropy function is via the "20 questions" game. This cooperative game is played by two players, Alice and Bob: Alice picks a distribution π over the numbers $\{1, \ldots, n\}$, and announces it to Bob. She then chooses a number x according to π, and Bob attempts to identify x using as few Yes/No queries as possible, on average.

An optimal strategy for the "20 questions" game is given by a Huffman code for π: Bob's questions reveal the codeword for x bit by bit. This strategy finds x using fewer than $H(\pi) + 1$ questions on average. However, the questions asked by Bob could be arbitrary. In this paper, we investigate the following question: *Are there restricted sets of questions that match the performance of Huffman codes, either exactly or approximately?*

Our main result gives a set $\mathcal{Q}$ of $1.25^{n+o(n)}$ questions such that for every distribution π, Bob can implement an *optimal* strategy for π using only questions from $\mathcal{Q}$. We also show that $1.25^{n-o(n)}$ allowed questions are needed, for infinitely many n. When allowing a small slack of r questions for identifying x over the optimal strategy, we show that a set of roughly $(rn)^{\Theta(1/r)}$ allowed questions is necessary and sufficient.

1 Introduction

A basic combinatorial and operational interpretation of Shannon's entropy function, which is often taught in introductory courses on information theory, is via the "20 questions" game (see for example the well-known textbook [6]). This game is played between two players, Alice and Bob: Alice picks a distribution π over a (finite) set of objects X, and announces it to Bob. Alice then chooses an object x according to π, and Bob attempts to identify the object using as few Yes/No queries as possible, on average. What questions should Bob ask? An optimal strategy for Bob is to compute a Huffman code for π, and then follow the corresponding decision tree: his first query, for example, asks whether x lies in the left subtree of the root. While this strategy minimizes the expected number of queries, the queries are arbitrary and can correspond to any of the $2^{|X|}$ subsets of $|X|$.

Motivated by this, we investigate the following meta-question, which guides this work: *How small can a set of allowed queries $\mathcal{Q}$ be such that for any distribution, there is a "high quality" strategy that uses only queries from $\mathcal{Q}$?*

[*]Electrical Engineering and Computer Science, Massachusetts Institute of Technology. Email: `yuvaldag@gmail.com`. The research was funded by ISF grant 1337/16.

[†]Computer Science Department, Technion — Israel Institute of Technology. Email: `yuvalfi@cs.technion.ac.il`. Taub Fellow — supported by the Taub Foundations. The research was funded by ISF grant 1337/16.

[‡]Zcash. Email: `ariel@z.cash`.

[§]School of Mathematics, Institute for Advanced Study. Email: `shaymoran1@gmail.com`. Research supported by the National Science Foundation under agreement No. CCF-1412958 and by the Simons Foundations.

Formalizing this question depends on how "high quality" is quantified. We consider the following combinatorial benchmark: A set of queries $\mathcal{Q}$ is *r-optimal* (or has *prolixity* r) if for every distribution π there is a strategy using queries from $\mathcal{Q}$ that finds x with at most $\mathrm{Opt}(\pi) + r$ queries on average when x is drawn according to π, where $\mathrm{Opt}(\pi)$ is the expected number of queries asked by an optimal strategy for π (e.g. a Huffman tree).

Our goal is to find the minimal size of an r-optimal set of queries, for given r.

We first address the case $r = 0$, in which we are searching for an allowed set of questions matching the performance of Huffman codes. Can the optimal performance of Huffman codes be achieved without using all possible queries? Our main result answers this in the affirmative:

Theorem (restatement of Theorem 3.2 and Theorem 3.3)**.** *For every n there is a set $\mathcal{Q}$ of $1.25^{n+o(n)}$ queries such that for every distribution over $[n]$, there is a strategy using only queries from $\mathcal{Q}$ which matches the performance of the optimal (unrestricted) strategy* exactly*. Furthermore, for infinitely many n, at least $1.25^{n-o(n)}$ queries are required to achieve this feat.*

One drawback of our construction is that it is randomized. Thus, we do not consider it particularly "efficient" nor "natural". It is interesting to find an explicit set $\mathcal{Q}$ that achieves this bound. Our best explicit construction is:

Theorem (restatement of Theorem 3.5)**.** *For every n there is an explicit set $\mathcal{Q}$ of $O(2^{n/2})$ queries such that for every distribution over $[n]$, there is a strategy using only queries from $\mathcal{Q}$ which matches the performance of the optimal (unrestricted) strategy* exactly*. Moreover, this strategy can be computed in time $O(n^2)$.*

We move to the case of $r > 0$. Let $u(n, r)$ denote the minimum size of an r-optimal set of queries. We show that for any fixed $r > 0$, significant savings can be achieved:

Theorem (restatement of Theorem 4.1)**.** *For all $r \in (0, 1)$:*

$$(rn)^{\frac{1}{4r}} \lesssim u(n,r) \lesssim (rn)^{\frac{16}{r}}.$$

Instead of the exponential number of questions needed to match Huffman's algorithm exactly, for fixed $r > 0$ an r-optimal set of questions has polynomial size. In this case the upper bound is achieved by an explicit set of queries $\mathcal{Q}_r$. We also present an efficient randomized algorithm for computing an r-optimal strategy that uses queries from $\mathcal{Q}_r$.

For larger r, we have the following result, which relies on a sequel to this work [7]:

Theorem (restatement of Theorem 4.3)**.** *For all $r \geq 1$:*

$$\frac{1}{e}\lfloor r+1 \rfloor n^{1/\lfloor r+1 \rfloor} \leq u(n,r) \leq 2\lfloor r \rfloor n^{1/\lfloor r \rfloor}.$$

Related work The "20 questions" game is the starting point of combinatorial search theory [17, 1, 3]. Combinatorial search theory considers many different variants of the game, such as several unknown elements, non-adaptive queries, non-binary queries, and a non-truthful Alice [27, 28, 2, 8]. Both average-case and worst-case complexity measures are of interest. An important topic in combinatorial search theory is combinatorial group testing [9, 10].

A fair amount of literature [12, 16, 5, 23, 21, 22] studies the *redundancy* of π, which is defined by $\mathrm{Opt}(\pi) - H(\pi)$, as a function of various parameters of π, such as the maximum probability of an element. One can also study the analog of $u(n, r)$ in which the quality of a set of questions depends on how the cost compares to $H(\pi)$ rather than to $\mathrm{Opt}(\pi)$, a route taken in the sequel to this work [7].

We are unaware of any prior work that has considered the quantity $u(n, r)$. However, several particular sets of questions have been analyzed in the literature from the perspective of redundancy or prolixity, such as binary search trees [13, 25, 26, 14] and comparison-based sorting algorithms [11, 24].

2 Preliminaries

Notation We use $\log n$ for the base 2 logarithm of n and $[n]$ to denote the set $\{1, \ldots, n\}$.

Throughout the paper, we will consider probability distributions over the set $X_n = \{x_1, \ldots, x_n\}$ of size n. If π is a probability distribution over X_n, we will denote the probability of x_i by π_i, and the probability of a set $S \subseteq X_n$ by $\pi(S)$.

Information theory We use $H(\pi)$ to denote the base 2 entropy of a distribution π and $D(\pi\|\mu)$ to denote the Kullback–Leibler divergence. The *binary entropy* function $h(p)$ is the entropy of a Bernoulli random variable with success probability p. When Y is a Bernoulli random variable, the chain rule takes the following form:

$$H(X, Y) = h(\Pr[Y = 1]) + \Pr[Y = 0]H(X|Y = 0) + \Pr[Y = 1]H(X|Y = 1).$$

We call this the *Bernoulli chain rule*.

Decision trees In this paper we consider the task of revealing a secret element x from X_n by using Yes/No questions. Such a strategy will be called a *decision tree* or an *algorithm*.

A decision tree is a binary tree in which the internal nodes are labeled by subsets of X_n (which we call *questions* or *queries*), each internal node has two outgoing edges labeled *Yes* (belongs to the question set) and *No* (doesn't belong to the question set), and each leaf in the tree is labeled by those elements of X_n that are consistent with the queries on the path towards the leaf. The depth of an element x_i in a decision tree T, denoted $T(x_i)$, is the number of edges in the unique path from the root to the unique leaf labeled x_i.

Decision trees can be thought of as annotated prefix codes: the code of an element x_i is the concatenation of the labels of the edges leading to it. The mapping can also be used in the other direction: each binary prefix code of cardinality n corresponds to a unique decision tree over X_n.

Given a set $\mathcal{Q} \subseteq 2^{X_n}$ (a set of *allowed questions*), a *decision tree using* $\mathcal{Q}$ is one in which all questions belong to $\mathcal{Q}$. A decision tree is *valid* for a distribution μ if no two elements in the support of μ are consistent with the same leaf, or equivalently, if each leaf is labeled by at most one element in the support of μ.

Given a distribution μ and a decision tree T valid for μ, the *cost* (or *query complexity*) of T on μ, labeled $T(\mu)$, is the average number of questions asked on a random element, $T(\mu) = \sum_{i=1}^{n} \mu_i T(x_i)$. Given a set $\mathcal{Q}$ of allowed questions and a distribution μ, the *optimal cost* of μ with respect to $\mathcal{Q}$, denoted $c(\mathcal{Q}, \mu)$, is the minimal cost of a valid decision tree for μ using $\mathcal{Q}$.

Dyadic distributions and Huffman's algorithm Huffman's algorithm [15] finds the optimal cost of an unrestricted decision tree for a given distribution:

$$\mathrm{Opt}(\mu) = c(2^{X_n}, \mu).$$

We call a decision tree with this cost a *Huffman tree*[1] or an *optimal decision tree* for μ. More generally, a decision tree is r*-optimal* for μ if its cost is at most $\mathrm{Opt}(\mu) + r$.

It will be useful to consider this definition from a different point of view. Say that a distribution is *dyadic* if the probability of every element is either 0 or of the form 2^{-d} for some integer d. We can associate with each decision tree T a distribution τ on the leaves of T given by $\tau_i = 2^{-T(x_i)}$. This gives a correspondence between decision trees and dyadic distributions.

In the language of dyadic distributions, Huffman's algorithm solves the following optimization problem:

$$\mathrm{Opt}(\mu) = \min_{\substack{\tau \text{ dyadic} \\ \mathrm{supp}(\tau)=\mathrm{supp}(\mu)}} \sum_{i=1}^{n} \mu_i \log \frac{1}{\tau_i} = \min_{\substack{\tau \text{ dyadic} \\ \mathrm{supp}(\tau)=\mathrm{supp}(\mu)}} \left[H(\mu) + D(\mu\|\tau) \right].$$

In other words, computing $\mathrm{Opt}(\mu)$ amounts to minimizing $D(\mu\|\tau)$, and thus to "rounding" μ to a dyadic distribution. We call τ a *Huffman distribution* for μ.

The following classical inequality shows that $\mathrm{Opt}(\mu)$ is very close to the entropy of μ:

$$H(\mu) \le \mathrm{Opt}(\mu) < H(\mu) + 1.$$

The lower bound follows from the non-negativity of the Kullback–Leibler divergence; it is tight exactly when μ is dyadic. The upper bound from the Shannon–Fano code, which corresponds to the dyadic sub-distribution $\tau_i = 2^{-\lceil \log \mu_i \rceil}$ (in which the probabilities could sum to less than 1).

Prolixity We measure the quality of sets of questions by comparing the cost of decision trees using them to the cost of optimal decision trees (for the difference we coin the term *prolixity*). In more detail, the *prolixity* of a decision tree T for a distribution μ is $T(\mu) - \mathrm{Opt}(\mu)$, and the prolixity of a set of questions $\mathcal{Q}$ is the supremum of $c(\mathcal{Q}, \mu) - \mathrm{Opt}(\mu)$ over all distributions μ on X_n.[2]

The parameter $u(n, r)$ Our main object of study in this paper are the parameters $u(n, r)$. The parameter $u(n, r)$ is the cardinality of the smallest set of questions $\mathcal{Q} \subseteq 2^{X_n}$ whose prolixity is at most r, that is, which satisfies $c(\mathcal{Q}, \mu) \le \mathrm{Opt}(\mu) + r$ for all distributions μ on X_n.

A useful lemma The following simple lemma will be used several times in the rest of the paper.

Lemma 2.1. *Let $p_1 \ge \ldots \ge p_n$ be a non-increasing list of numbers of the form $p_i = 2^{-a_i}$ (for integer a_i), and let $a \le a_1$ be an integer. If $\sum_{i=1}^{n} p_i \ge 2^{-a}$ then for some m we have $\sum_{i=1}^{m} p_i = 2^{-a}$.*

If furthermore $\sum_{i=1}^{n} p_i$ is a multiple of 2^{-a} then for some ℓ we have $\sum_{i=\ell}^{n} p_i = 2^{-a}$.

Proof. Let m be the maximal index such that $\sum_{i=1}^{m} p_i \le 2^{-a}$. If $m = n$ then we are done, so suppose that $m < n$. Let $S = \sum_{i=1}^{m} p_i$. We would like to show that $S = 2^{-a}$.

The condition $p_1 \le \cdots \le p_n$ implies that $a_{m+1} \ge \cdots \ge a_1$, and so $k := 2^{a_{m+1}} S = \sum_{i=1}^{m} 2^{a_{m+1} - a_i}$ is an integer. By assumption $k \le 2^{a_{m+1} - a}$ whereas $k + 1 = 2^{a_{m+1}} \sum_{i=1}^{m+1} p_i > 2^{a_{m+1} - a}$. Since $2^{a_{m+1} - a}$ is itself an integer (since $a_{m+1} \ge a_1 \ge a$), we conclude that $k = 2^{a_{m+1} - a}$, and so $S = 2^{-a}$.

To prove the furthermore part, notice that by repeated applications of the of the lemma we can partition $[n]$ into intervals whose probabilities are 2^{-a}. The last such interval provides the required index ℓ. $\square$

[1] This is a slight abuse of notation, since there exist optimal trees that cannot be produced by Huffman's algorithm [12].

[2] This should be distinguished from the more common *redundancy*, which measures the difference between the cost of the decision tree and the *entropy* of the distribution, that is, $T(\mu) - H(\mu)$.

3 Zero prolixity

In this section we investigate the minimal size of a set of allowed questions which matches the performance of minimum redundancy codes such as Huffman codes.

Let us repeat the definition of an optimal set of questions that is central in this section.

Definition 3.1. A set $\mathcal{Q}$ of subsets of X_n is an *optimal set of questions over* X_n if for all distributions μ on X_n,

$$c(\mathcal{Q}, \mu) = \mathrm{Opt}(\mu).$$

Using the above definition, $u(n, 0)$ is equal to the minimal size of an optimal set of questions over X_n. Perhaps surprisingly, the trivial upper bound of 2^{n-1} on $u(n, 0)$ can be exponentially improved:

Theorem 3.2. *We have*

$$u(n, 0) \leq 1.25^{n+o(n)}.$$

We prove a similar lower bound, which is almost tight for infinitely many n:

Theorem 3.3. *For n of the form $n = 5 \cdot 2^m$,*

$$u(n, 0) \geq 1.25^n / O(\sqrt{n}).$$

For all n,

$$u(n, 0) \geq 1.232^n / O(\sqrt{n}).$$

Corollary 3.4. *We have*

$$\limsup_{n \to \infty} \frac{\log u(n, 0)}{n} = \log 1.25.$$

Unfortunately, the construction in Theorem 3.2 is not explicit. A different construction, which uses $O(\sqrt{2}^n)$ questions, is not only explicit, but can also be implemented efficiently:

Theorem 3.5. *Consider the set of questions*

$$\mathcal{Q} = \{A \subseteq X_n : A \subseteq X_{\lceil n/2 \rceil} \text{ or } A \supseteq X_{\lceil n/2 \rceil}\}.$$

The set $\mathcal{Q}$ consists of $2^{\lceil n/2 \rceil} + 2^{\lfloor n/2 \rfloor}$ questions and satisfies the following properties:

1. *There is an indexing scheme $\mathcal{Q} = \{Q_q : q \in \{0, 1\}^{\lceil n/2 \rceil + 1}\}$ such that given an index q and an element $x_i \in X_n$, we can decide whether $x_i \in Q_q$ in time $O(n)$.*

2. *Given a distribution π, we can construct an optimal decision tree for π using $\mathcal{Q}$ in time $O(n^2)$.*

3. *Given a distribution π, we can implement an optimal decision tree for π in an online fashion in time $O(n)$ per question, after $O(n \log n)$ preprocessing.*

Section organization. Section 3.1 shows that a set of questions is optimal if and only if it is a *dyadic hitter*, that is, contains a question splitting every non-constant dyadic distribution into two equal halves. Section 3.2 discusses a relation to hitting sets for maximal antichains, and proves Theorem 3.5. Section 3.3 shows that the optimal size of a dyadic hitter is controlled by the minimum value of another parameter, the *maximum relative density*. We upper bound the minimum value in Section 3.4, thus proving Theorem 3.3, and lower bound it in Section 3.5, thus proving Theorem 3.2.

3.1 Reduction to dyadic hitters

The purpose of this subsection is to give a convenient combinatorial characterization of optimal sets of questions. Before presenting this characterization, we show that in this context it suffices to look at dyadic distributions.

Lemma 3.1.1. *A set $\mathcal{Q}$ of questions over X_n is optimal if and only if $c(\mathcal{Q},\mu) = \mathrm{Opt}(\mu)$ for all* dyadic *distributions μ.*

Proof. Suppose that $\mathcal{Q}$ is optimal for all dyadic distributions, and let π be an arbitrary distribution over X_n. Let μ be a dyadic distribution such that

$$\mathrm{Opt}(\pi) = \sum_{i=1}^{n} \pi_i \log \frac{1}{\mu_i}.$$

By assumption, $\mathcal{Q}$ is optimal for μ. Let T be an optimal decision tree for μ using questions from $\mathcal{Q}$ only, and let τ be the corresponding dyadic distribution, given by $\tau_i = 2^{-T(x_i)}$ (recall that $T(x_i)$ is the depth of x_i). Since τ minimizes $T(\mu) = H(\mu) + D(\mu\|\tau)$ over dyadic distributions, necessarily $\tau = \mu$. Thus

$$T(\pi) = \sum_{i=1}^{n} \pi_i \log \frac{1}{\tau_i} = \sum_{i=1}^{n} \pi_i \log \frac{1}{\mu_i} = \mathrm{Opt}(\pi),$$

showing that $\mathcal{Q}$ is optimal for μ. □

Given a dyadic distribution μ on X_n, we will be particularly interested in the collection of subsets of X_n that have probability exactly half under μ.

Definition 3.1.2 (Dyadic hitters)**.** Let μ be a non-constant dyadic distribution. A set $A \subseteq X_n$ *splits* μ if $\mu(A) = 1/2$. We denote the collection of all sets splitting μ by $\mathrm{Spl}(\mu)$. We call a set of the form $\mathrm{Spl}(\mu)$ a *dyadic set*.

We call a set of questions $\mathcal{Q}$ a *dyadic hitter* in X_n if it intersects $\mathrm{Spl}(\mu)$ for all non-constant dyadic distributions μ. (Lemma 2.1 implies that $\mathrm{Spl}(\mu)$ is always non-empty.)

A dyadic hitter is precisely the object we are interested in:

Theorem 3.1.1. *A set $\mathcal{Q}$ of subsets of X_n is an optimal set of questions if and only if it is a dyadic hitter in X_n.*

Proof. Let $\mathcal{Q}$ be a dyadic hitter in X_n. We prove by induction on $1 \le m \le n$ that for a dyadic distribution μ on X_n with support size m, $c(\mathcal{Q},\mu) = H(\mu)$. Since $\mathrm{Opt}(\mu) = H(\mu)$, Lemma 3.1.1 implies that $\mathcal{Q}$ is an optimal set of questions.

The base case, $m = 1$, is trivial. Suppose therefore that μ is a dyadic distribution whose support has size $m > 1$. In particular, μ is not constant, and so $\mathcal{Q}$ contains some set $S \in \mathrm{Spl}(\mu)$. Let $\alpha = \mu|_S$ and $\beta = \mu|_{\overline{S}}$, and note that α, β are both dyadic. The induction hypothesis shows that $c(\mathcal{Q},\alpha) = H(\alpha)$ and $c(\mathcal{Q},\beta) = H(\beta)$. A decision tree which first queries S and then uses the implied algorithms for α and β has cost

$$1 + \frac{1}{2}H(\alpha) + \frac{1}{2}H(\beta) = h(\mu(S)) + \mu(S)H(\mu|_S) + \mu(\overline{S})H(\mu|_{\overline{S}}) = H(\mu),$$

using the Bernoulli chain rule; here $\mu|_S$ is the restriction of μ to the elements in S.

Conversely, suppose that $\mathcal{Q}$ is not a dyadic hitter, and let μ be a non-constant dyadic distribution such that $\mathrm{Spl}(\mu)$ is disjoint from $\mathcal{Q}$. Let T be any decision tree for μ using $\mathcal{Q}$, and let S be its first question. The cost of T is at least

$$1 + \mu(S)H(\mu|_S) + \mu(\overline{S})H(\mu|_{\overline{S}}) > h(\mu(S)) + \mu(S)H(\mu|_S) + \mu(\overline{S})H(\mu|_{\overline{S}}) = H(\mu),$$

since $\mu(S) \neq \frac{1}{2}$. Thus $c(\mathcal{Q}, \mu) > \mathrm{Opt}(\mu)$, and so $\mathcal{Q}$ is not an optimal set of questions. $\square$

3.2 Dyadic sets as antichains

There is a surprising connection between dyadic hitters and hitting sets for maximal antichains. We start by defining the latter:

Definition 3.2.1. A *fibre* in X_n is a subset of 2^{X_n} which intersects every maximal antichain in X_n.

Fibres were defined by Lonc and Rival [20], who also gave a simple construction, via cones:

Definition 3.2.2. The *cone* $\mathfrak{C}(S)$ of a set S consists of all subsets and all supersets of S.

Any cone $\mathfrak{C}(S)$ intersects any maximal antichain A, since otherwise $A \cup \{S\}$ is also an antichain. By choosing S of size $\lfloor n/2 \rfloor$, we obtain a fibre of size $2^{\lfloor n/2 \rfloor} + 2^{\lceil n/2 \rceil} - 1 = \Theta(2^{n/2})$. Our goal now is to show that every fibre is a dyadic hitter:

Theorem 3.2.3. *every fibre is a dyadic hitter.*

This shows that every cone is a dyadic hitter, and allows us to give a simple algorithm for constructing an optimal decision tree using any cone.

We start with a simple technical lemma that will also be used in Section 3.4:

Definition 3.2.4. Let μ be a dyadic distribution over X_n. The *tail* of μ is the largest set of elements $T \subseteq X_n$ such that for some $a \geq 1$,

(i) The elements in T have probabilities $2^{-a-1}, 2^{-a-2}, \ldots, 2^{-a-(|T|-1)}, 2^{-a-(|T|-1)}$.

(ii) Every element not in T has probability at least 2^{-a}.

Lemma 3.2.5. *Suppose that μ is a non-constant dyadic distribution with non-empty tail T. Every set in $\mathrm{Spl}(\mu)$ either contains T or is disjoint from T.*

Proof. The proof is by induction on $|T|$. If $|T| = 2$ then there exist an integer $a \geq 1$ and two elements, without loss of generality x_1, x_2, of probability 2^{-a-1}, such that all other elements have probability at least 2^{-a}. Suppose that $S \in \mathrm{Spl}(\mu)$ contains exactly one of x_1, x_2. Then

$$2^{a-1} = \sum_{x_i \in S} 2^a \mu(x_i) = \sum_{x_i \in S \setminus \{x_1, x_2\}} 2^a \mu(x_i) + \frac{1}{2}.$$

However, the left-hand side is an integer while the right-hand side is not. We conclude that S must contain either both of x_1, x_2 or none of them.

For the induction step, let the elements in the tail T of μ have probabilities $2^{-a-1}, 2^{-a-2}, \ldots, 2^{-a-(|T|-1)}, 2^{-a-(|T|-1)}$. Without loss of generality, suppose that x_{n-1}, x_n are the elements whose probability is $2^{-a-(|T|-1)}$. The same argument as before shows that every dyadic set in $\mathrm{Spl}(\mu)$ must contain either both of x_{n-1}, x_n or neither. Form a new dyadic distribution ν on X_{n-1} by merging the elements x_{n-1}, x_n into x_{n-1}, and note that $\mathrm{Spl}(\mu)$ can be obtained from $\mathrm{Spl}(\nu)$ by replacing x_{n-1} with x_{n-1}, x_n. The distribution ν has tail $T' = T \setminus \{x_n\}$, and so by induction, every set in $\mathrm{Spl}(\nu)$ either contains T' or is disjoint from T'. This implies that every set in $\mathrm{Spl}(\mu)$ either contains T or is disjoint from T. $\square$

The first step in proving Theorem 3.2.3 is a reduction to dyadic distributions having full support:

Lemma 3.2.6. *A set of questions is a dyadic hitter in X_n if and only if it intersects* $\mathrm{Spl}(\mu)$ *for all non-constant full-support dyadic distributions μ on X_n.*

Proof. A dyadic hitter clearly intersects $\mathrm{Spl}(\mu)$ for all non-constant full-support dyadic distributions on X_n. For the other direction, suppose that $\mathcal{Q}$ is a set of questions that intersects $\mathrm{Spl}(\mu)$ for every non-constant full-support dyadic distribution μ. Let ν be a non-constant dyadic distribution on X_n which doesn't have full support. Let $x_{\min}$ be an element in the support of ν with minimal probability, which we denote $\nu_{\min}$. Arrange the elements in $\overline{\mathrm{supp}(\nu)}$ in some arbitrary order $x_{i_1}, \ldots, x_{i_m}$. Consider the distribution μ given by:

- $\mu(x_i) = \nu(x_i)$ if $x_i \in \mathrm{supp}(\mu)$ and $x_i \neq x_{\min}$.
- $\mu(x_{\min}) = \nu_{\min}/2$.
- $\mu(x_{i_j}) = \nu_{\min}/2^{j+1}$ for $j < m$.
- $\mu(x_{i_m}) = \nu_{\min}/2^m$.

In short, we have replaced $\nu(x_{\min}) = \nu_{\min}$ with a tail $x_{\min}, x_{i_1}, \ldots, x_{i_m}$ of the same total probability. It is not hard to check that μ is a non-constant dyadic distribution having full support on X_n.

We complete the proof by showing that $\mathcal{Q}$ intersects $\mathrm{Spl}(\nu)$. By assumption, $\mathcal{Q}$ intersects $\mathrm{Spl}(\mu)$, say at a set S. Lemma 3.2.5 shows that S either contains all of $\{x_{\min}\} \cup \overline{\mathrm{supp}(\nu)}$, or none of these elements. In both cases, $\nu(S) = \mu(S) = 1/2$, and so $\mathcal{Q}$ intersects $\mathrm{Spl}(\nu)$. $\square$

We complete the proof of Theorem 3.2.3 by showing that dyadic sets corresponding to full-support distributions are maximal antichains:

Lemma 3.2.7. *Let μ be a non-constant dyadic distribution over X_n with full support, and let $D = \mathrm{Spl}(\mu)$. Then D is a maximal antichain which is closed under complementation (i.e. $A \in D \implies X \setminus A \in D$).*

Proof. (i) That D is closed under complementation follows since if $A \in D$ then $\mu(X \setminus A) = 1 - \mu(A) = 1/2$.

(ii) That D is an antichain follows since if A strictly contains B then $\mu(A) > \mu(B)$ (because μ has full support).

(iii) It remains to show that D is maximal. By (i) it suffices to show that every B with $\mu(B) > 1/2$ contains some $A \in D$. This follows from applying Lemma 2.1 on the probabilities of the elements in B. $\square$

Cones allow us to prove Theorem 3.5:

Proof of Theorem 3.5. Let $S = \{x_1, \ldots, x_{\lfloor n/2 \rfloor}\}$. The set of questions $\mathcal{Q}$ is the cone $\mathfrak{C}(S)$, whose size is $2^{\lfloor n/2 \rfloor} + 2^{\lceil n/2 \rceil} - 1 < 2^{\lceil n/2 \rceil + 1}$.

An efficient indexing scheme for $\mathcal{Q}$ divides the index into a bit b, signifying whether the set is a subset of S or a superset of S, and $\lfloor n/2 \rfloor$ bits (in the first case) or $\lceil n/2 \rceil$ bits (in the second case) for specifying the subset or superset.

To prove the other two parts, we first solve an easier question. Suppose that μ is a non-constant dyadic distribution whose sorted order is known. We show how to find a set in $\mathrm{Spl}(\mu) \cap \mathcal{Q}$ in time $O(n)$. If $\mu(S) = 1/2$ then $S \in \mathrm{Spl}(\mu)$. If $\mu(S) > 1/2$, go over the elements in S in non-decreasing order. According to Lemma 2.1, some prefix will sum to $1/2$ exactly. If $\mu(S) < 1/2$, we do the same with $\overline{S}$, and then complement the result.

Suppose now that π is a non-constant distribution. We can find a Huffman distribution μ for π and compute the sorted order of π in time $O(n \log n)$. The second and third part now follow as in the proof of Theorem 3.1.1. □

3.3 Reduction to maximum relative density

Our lower bound on the size of a dyadic hitter, proved in the following subsection, will be along the following lines. For appropriate values of n, we describe a dyadic distribution μ, all of whose splitters have a certain size i or $n - i$. Moreover, only a ρ fraction of sets of size i split μ. We then consider all possible permutations of μ. Each set of size i splits a ρ fraction of these, and so any dyadic hitter must contain at least $1/\rho$ sets.

This lower bound argument prompts the definition of *maximum relative density* (MRD), which corresponds to the parameter ρ above; in the general case we will also need to optimize over i. We think of the MRD as a property of dyadic sets rather than dyadic distributions; indeed, the concept of MRD makes sense for any collection of subsets of X_n. If a dyadic set has MRD ρ then any dyadic hitter must contain at least $1/\rho$ questions, due to the argument outlined above. Conversely, using the probabilistic method we will show that roughly $1/\rho_{\min}(n)$ questions suffice, where $\rho_{\min}(n)$ is the minimum MRD of a dyadic set on X_n.

Definition 3.3.1 (Maximum relative density)**.** Let D be a collection of subsets of X_n. For $0 \leq i \leq n$, let

$$\rho_i(D) := \frac{\left|\left\{S \in D : |S| = i\right\}\right|}{\binom{n}{i}}.$$

We define the *maximum relative density* (MRD) of D, denoted $\rho(D)$, as

$$\rho(D) := \max_{i \in \{1,\ldots,n-1\}} \rho_i(D).$$

We define $\rho_{\min}(n)$ to be the minimum of $\rho(D)$ over all dyadic sets. That is, $\rho_{\min}(n)$ is the smallest possible maximum relative density of a set of the form $\mathrm{Spl}(\mu)$.

The following theorem shows that $u(n, 0)$ is controlled by $\rho_{\min}(n)$, up to polynomial factors.

Theorem 3.3.1. *Fix an integer n, and denote $M := \frac{1}{\rho_{\min}(n)}$. Then*

$$M \leq u(n, 0) \leq n^2 \log n \cdot M.$$

Proof. Note first that according to Theorem 3.1.1, $u(n, 0)$ is equal to the minimal size of a dyadic hitter in X_n, and thus it suffices to provide the corresponding lower and upper bounds for this size.

Let σ be a uniformly random permutation on X_n. If S is any set of size i then $\sigma^{-1}(S)$ is a uniformly random set of size i, and so

$$\rho_i(D) = \Pr_{\sigma \in \mathrm{Sym}(X_n)}[\sigma^{-1}(S) \in D] = \Pr_{\sigma \in \mathrm{Sym}(X_n)}[S \in \sigma(D)].$$

(Here $\mathrm{Sym}(X_n)$ is the group of permutations of X_n.)

Fix a dyadic set D on X_n with $\rho(D) = \rho_{\min}(n)$. The formula for $\rho_i(D)$ implies that for any subset S of X_n (of *any* size),

$$\Pr_{\sigma \in \mathrm{Sym}(X_n)}[S \in \sigma(D)] \leq \rho_{\min}(n).$$

Let $\mathcal{Q}$ be a collection of subsets of X_n with $|\mathcal{Q}| < M$. A union bound shows that

$$\Pr_{\sigma \in \mathrm{Sym}(X_n)}[\mathcal{Q} \cap \sigma(D) \neq \emptyset] \leq |\mathcal{Q}| \rho_{\min}(n) < 1.$$

Thus, there exists a permutation σ such that $\mathcal{Q} \cap \sigma(D) = \emptyset$. Since $\sigma(D)$ is also a dyadic set, this shows that $\mathcal{Q}$ is not a dyadic hitter. We deduce that any dyadic hitter must contain at least M questions.

For the upper bound on $u(n,0)$, construct a set of subsets $\mathcal{Q}$ containing, for each $i \in \{1, \ldots, n-1\}$, $Mn \log n$ uniformly chosen sets $S \subseteq X_n$ of size i. We show that with positive probability, $\mathcal{Q}$ is a dyadic hitter.

Fix any dyadic set D, and let $i \in \{1, \ldots, n-1\}$ be such that $\rho_i(D) = \rho(D) \geq \rho_{\min}(n)$. The probability that a random set of size i doesn't belong to D is at most $1 - \rho(D) \leq 1 - \rho_{\min}(n)$. Therefore the probability that $\mathcal{Q}$ is disjoint from D is at most

$$(1 - \rho_{\min}(n))^{Mn \log n} \leq e^{-\rho_{\min}(n) Mn \log n} = e^{-n \log n} < n^{-n}.$$

As we show below in Claim 3.3.2, there are at most n^n non-constant dyadic distributions, and so a union bound implies that with positive probability, $\mathcal{Q}$ is indeed a dyadic hitter. □

In order to complete the proof of Theorem 3.3.1, we bound the number of non-constant dyadic distributions:

Claim 3.3.2. *There are at most n^n non-constant dyadic distributions on X_n.*

Proof. Recall that dyadic distributions correspond to decision trees in which an element of probability $2^{-\ell}$ is a leaf at depth ℓ. Clearly the maximal depth of a leaf is $n-1$, and so the probability of each element in a non-constant dyadic distribution is one of the n values $0, 2^{-1}, \ldots, 2^{-(n-1)}$. The claim immediately follows. □

Krenn and Wagner [19] showed that the number of full-support dyadic distributions on X_n is asymptotic to $\alpha \gamma^{n-1} n!$, where $\alpha \approx 0.296$ and $\gamma \approx 1.193$, implying that the number of dyadic distributions on X_n is asymptotic to $\alpha e^{1/\gamma} \gamma^{n-1} n!$. Boyd [4] showed that the number of monotone full-support dyadic distributions on X_n is asymptotic to $\beta \lambda^n$, where $\beta \approx 0.142$ and $\lambda \approx 1.794$, implying that the number of monotone dyadic distributions on X_n is asymptotic to $\beta(1+\lambda)^n$.

The proof of Theorem 3.3.1 made use of two properties of dyadic sets:

1. Any permutation of a dyadic set is a dyadic set.
2. There are $e^{n^{O(1)}}$ dyadic sets.

If $\mathcal{F}$ is any collection of subsets of 2^{X_n} satisfying the first property then the proof of Theorem 3.3.1 generalizes to show that the minimal size U of a hitting set for $\mathcal{F}$ satisfies

$$M \leq U \leq Mn \log |\mathcal{F}|, \qquad \text{where } M = \frac{1}{\min_{D \in \mathcal{F}} \rho(D)}.$$

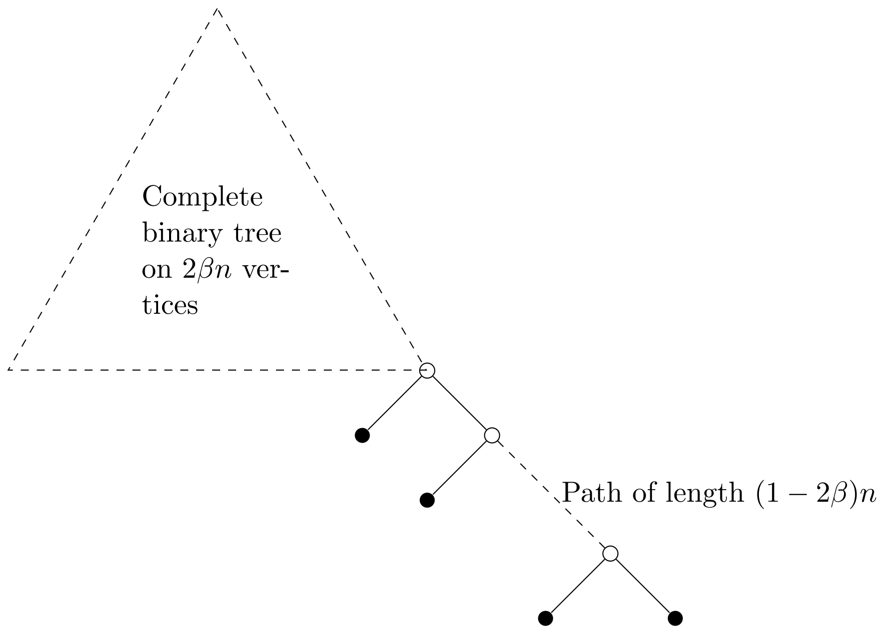

Figure 1: The hard distribution used to prove Lemma 3.4.1, in decision tree form

3.4 Upper bounding $\rho_{\min}(n)$

Theorem 3.3 will ultimately follow from the following lemma, by way of Theorem 3.3.1:

Lemma 3.4.1. *Fix $0 < \beta \leq 1/2$. There exists an infinite sequence of positive integers n (namely, those of the form $\lfloor \frac{2^a}{2\beta} \rfloor$ for integer a) such that some dyadic set D in X_n satisfies $\rho(D) \leq 7\sqrt{n}2^{-(h(\beta)-2\beta)n}$.*

Proof. We prove the lemma under the simplifying assumption that $1/\beta$ is an integer (our most important application of the lemma has $\beta := 1/5$). Extending the argument for general β is straightforward and left to the reader.

Let n be an integer of the form $\frac{2^a}{2\beta}$, for a positive integer a. Note that for n of this form, $\beta n = 2^{a-1}$ is a power of two. Let $t = \beta n$, and construct a dyadic distribution μ on X_n as follows:

1. For $i \in [2t-1]$, $\mu(x_i) = 2^{-a} = \frac{1}{2t}$.
2. For $i \in [n-1] \setminus [2t-1]$, $\mu(x_i) = \mu(x_{i-1})/2 = 2^{-(a+i-2t+1)}$.
3. $\mu(x_n) = \mu(x_{n-1})$.

The corresponding decision tree is obtained by taking a complete binary tree of depth a and replacing one of the leaves by a "path" of length $n - 2^a$; see Figure 1. Alternatively, in the terminology of Definition 3.2.4 we form μ by taking the uniform distribution on X_{2t} and replacing x_{2t} with a tail on $x_{2t}, \ldots, x_n$.

We claim that $D := \mathrm{Spl}(\mu)$ contains only two types of sets:

1. Subsets of size t of X_{2t-1}.
2. Subsets of size $n-t$ containing $t-1$ elements of X_{2t-1} and all the elements $x_{2t}, \ldots, x_n$.

It is immediate that any such set S is in D. On the other hand, Lemma 3.2.5 shows that every set $S \in D$ either contains the tail $x_{2t}, \ldots, x_n$ or is disjoint from it. If S is disjoint from the tail then it must be of the first form, and if S contains the tail then it must be of the second form.

Using the estimate $\binom{n}{\beta n} \geq 2^{h(\beta)n}/e\sqrt{2\pi\beta n} \geq 2^{h(\beta)n}/7\sqrt{n}$ (see for example [18, Lemma 12]), we see that

$$\rho_t(D) = \rho_{n-t}(D) = \frac{\binom{2t-1}{t}}{\binom{n}{t}} \leq \frac{2^{2t}}{\binom{n}{\beta n}} \leq 7\sqrt{n}\frac{2^{2t}}{2^{h(\beta)n}} = 7\sqrt{n}2^{(2\beta-h(\beta))n}.$$

For $i \in \{1, \ldots, n-1\} \setminus \{t, n-t\}$ we have $\rho_i(D) = 0$. Thus indeed

$$\rho(D) \leq 7\sqrt{n}2^{(2\beta-h(\beta))n}. \qquad \square$$

Theorem 3.3 can now be easily derived. The first step is determining the optimal value of β:

Claim 3.4.2. *We have*

$$\max_{\beta\in[0,1]} 2^{h(\beta)-2\beta} = 1.25,$$

and the maximum is attained (uniquely) at $\beta = 1/5$.

Proof. Let $f(\beta) = h(\beta) - 2\beta$. The derivative $f'(\beta)$ is equal to

$$f'(\beta) = \log\left(\frac{1-\beta}{\beta}\right) - 2,$$

which is decreasing for $0 < \beta < 1$ and vanishes at $\beta = 1/5$. Thus $f(\beta)$ achieves a unique maximum over $\beta \in (0,1)$ at $\beta = 1/5$, where

$$2^{f(1/5)} = 2^{h(1/5)-2\cdot 1/5} = 1.25. \qquad \square$$

Proof of Theorem 3.3:

Proof. Let $\beta := 1/5$. Claim 3.4.2 shows that $2^{-(2\beta-h(\beta))} = 1.25$. Fix any n of the form $n = \frac{2^a}{2\beta}$ for a positive integer a. It follows from Lemma 3.4.1 together with the first inequality in Theorem 3.3.1 that $u(n,0) \geq 1.25^n/7\sqrt{n}$.

A general n can be written in the form $n = \frac{2^a}{2\beta}$ for a positive integer a and $1/4 \leq \beta \leq 1/2$. Lemma 3.4.1 and Theorem 3.3.1 show that for any integer $\ell \geq 0$,

$$u(n,0) \geq 2^{[h(\beta/2^\ell)-2\beta/2^\ell]n}/7\sqrt{n}.$$

Calculation shows that when $\beta \leq \beta_0 \approx 0.27052059413118146$, this is maximized at $\ell = 0$, and otherwise this is maximized at $\ell = 1$. Denote the resulting lower bound by $L(\beta)^n/7\sqrt{n}$, the minimum of $L(\beta)$ is attained at β_0, at which point its value is $L(\beta_0) \approx 1.23214280723432$. $\square$

3.5 Lower bounding $\rho_{\min}(n)$

We will derive Theorem 3.2 from the following lemma:

Lemma 3.5.1. *For every non-constant dyadic distribution μ there exists $0 < \beta < 1$ such that*

$$\rho(\mathrm{Spl}(\mu)) \geq \frac{2^{(2\beta-h(\beta))n}}{(7\sqrt{n})^{O(\log n)}} = 2^{(2\beta-h(\beta))n-o(n)}.$$

Proof. Assume without loss of generality that the probabilities in μ are non-increasing:

$$\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n.$$

The idea is to find a partition of X_n of the form

$$X_n = \bigcup_{i=1}^{\gamma} (D_i \cup E_i)$$

which satisfies the following properties:

1. D_i consists of elements having the same probability p_i.
2. If D_i has an even number of elements then $E_i = \emptyset$.
3. If D_i has an odd number of elements then $\mu(E_i) = p_i$.
4. $\gamma = O(\log n)$. (In fact, $\gamma = o(n/\log n)$ would suffice.)

We will show later how to construct such a partition.

The conditions imply that $\mu(D_i \cup E_i)$ is an even integer multiple of p_i, say $\mu(D_i \cup E_i) = 2c_i p_i$. It is not hard to check that $c_i = \lceil |D_i|/2 \rceil$.

Given such a partition, we show how to lower bound the maximum relative density of $\mathrm{Spl}(\mu)$. If $S_i \subseteq D_i$ is a set of size c_i for each $i \in [\gamma]$ then the set $S = \bigcup_i S_i$ splits μ:

$$\mu(S) = \sum_{i=1}^{\gamma} c_i p_i = \frac{1}{2} \sum_{i=1}^{\gamma} \mu(D_i \cup E_i) = \frac{1}{2}.$$

Defining $c = \sum_{i=1}^{\gamma} c_i$, we see that each such set S contains c elements, and the number of such sets is

$$\prod_{i=1}^{\gamma} \binom{|D_i|}{c_i} \geq \prod_{i=1}^{\gamma} \frac{2^{2c_i}}{(7\sqrt{n})} = \frac{2^{2c}}{(7\sqrt{n})^{O(\log n)}},$$

using the estimate

$$\binom{m}{\lceil m/2 \rceil} = \Theta\left(\frac{2^{2\lceil m/2 \rceil}}{\sqrt{m}}\right),$$

which follows from Stirling's approximation.

In order to obtain an estimate on the maximum relative density of $\mathrm{Spl}(\mu)$, we use the following folklore upper bound[3] on $\binom{n}{c}$:

$$\binom{n}{c} \leq 2^{h(c/n)n}.$$

We conclude that the maximum relative density of $\mathrm{Spl}(\mu)$ is at least

$$\rho(\mu) \geq \rho_c(\mu) \geq \frac{\prod_{i=1}^{\gamma} \binom{|D_i|}{c_i}}{\binom{n}{c}} \geq \frac{2^{2c - h(c/n)n}}{(7\sqrt{n})^{O(\log n)}}.$$

To obtain the expression in the statement of the lemma, take $\beta := c/n$.

We now show how to construct the partition of X_n. We first explain the idea behind the construction, and then provide full details; the reader who is interested only in the construction itself can skip ahead.

[3]Here is a quick proof: Let Y be a uniformly random subset of X_n of size c, and let Y_i indicate the event $x_i \in Y$. Then $\log \binom{n}{c} = H(Y) \leq nH(Y_1) = nh(c/n)$.

Proof idea Let $q_1, \dots, q_\gamma$ be the different probabilities of elements in μ. We would like to put all elements of probability q_i in the set D_i, but there are two difficulties:

1. There might be an odd number of elements whose probability is q_i.

2. There might be too many distinct probabilities, that is, γ could be too large. (We need $\gamma = o(n/\log n)$ for the argument to work.)

The second difficulty is easy to solve: we let $D_1 = \{x_1\}$, and use Lemma 2.1 to find an index ℓ such that $\mu(E_1) := \mu(\{x_\ell, \dots, x_n\}) = \mu_1$. A simple argument shows that all remaining elements have probability at least μ_1/n, and so the number of remaining distinct probabilities is $O(\log n)$. (The reader should observe the resemblance between E_1 and the tail of the hard distribution constructed in Lemma 3.4.1.)

Lemma 2.1 also allows us to resolve the first difficulty. The idea is as follows. Suppose that the current set under construction, D_i, has an odd number of elements, each of probability q_i. We use Lemma 2.1 to find a set of elements whose total probability is q_i, and put them in E_i.

Detailed proof Let N be the maximal index such that $\mu_N > 0$. Since μ is non-constant, $\mu_1 \le 1/2$, and so Lemma 2.1 proves the existence of an index M such that $\mu(\{x_{M+1}, \dots, x_N\}) = \mu_1$ (we use the *furthermore* part of the lemma, and $M = \ell - 1$). We take

$$D_1 := \{x_1\}, \quad E_1 := \{x_{M+1}, \dots, x_n\}.$$

Thus $\mu(D_1) = \mu(E_1) = \mu_1$, and so $\mu(\{x_2, \dots, x_M\}) = 1 - 2\mu_1$ (possibly $M = 1$, in which case the construction is complete).

By construction $n\mu_M > \mu(E_1) = \mu_1$, and so $\mu_M < \mu_1/n$. In particular, the number of distinct probabilities among $\mu_2, \dots, \mu_M$ is at most $\log n$. This will guarantee that $\gamma \le \log n + 1$, as will be evident from the construction.

The construction now proceeds in steps. At step i, we construct the sets D_i and E_i, given the set of available elements $\{x_{\alpha_i}, \dots, x_M\}$, where possibly $\alpha_i = M + 1$; in the latter case, we have completed the construction. We will maintain the invariant that $\mu(\{x_{\alpha_i}, \dots, x_M\})$ is an even multiple of μ_{α_i}; initially $\alpha_2 := 2$, and $\mu(\{x_{\alpha_i}, \dots, x_M\}) = (1/\mu_1 - 2)\mu_1$ is indeed an even multiple of μ_2.

Let β_i be the maximal index such that $\mu_{\beta_i} = \mu_{\alpha_i}$ (possibly $\beta_i = \alpha_i$). We define

$$D_i := \{x_{\alpha_i}, \dots, x_{\beta_i}\}.$$

Suppose first that $|D_i|$ is even. In this case we define $E_i := \emptyset$, and $\alpha_{i+1} := \beta_i + 1$. Note that

$$\mu(\{x_{\alpha_{i+1}}, \dots, x_M\}) = \mu(\{x_{\alpha_i}, \dots, x_M\}) - |D_i|\mu_{\alpha_i},$$

and so the invariant is maintained.

Suppose next that $|D_i|$ is odd. In this case $\mu(\{x_{\beta_i+1}, \dots, x_M\}) \ge x_{\alpha_i}$, since $\mu(\{x_{\beta_i+1}, \dots, x_M\})$ is an odd multiple of μ_{α_i}. Therefore we can use Lemma 2.1 to find an index γ_i such that $\mu(\{x_{\beta_i+1}, \dots, x_{\gamma_i}\}) = \mu_{\alpha_i}$. We take

$$E_i := \{x_{\beta_i+1}, \dots, x_{\gamma_i}\}$$

and $\alpha_{i+1} := \gamma_i + 1$. Note that

$$\mu(\{x_{\alpha_{i+1}}, \dots, x_M\}) = \mu(\{x_{\alpha_i}, \dots, x_M\}) - (|D_i| + 1)\mu_{\alpha_i},$$

and so the invariant is maintained.

The construction eventually terminates, say after step γ. The construction ensures that $\mu_{\alpha_2} > \mu_{\alpha_3} > \cdots > \mu_{\alpha_\gamma}$. Since there are at most $\log n$ distinct probabilities among the elements $\{x_{\alpha_2}, \ldots, x_M\}$, $\gamma \leq \log n + 1$, completing the proof. $\square$

Theorem 3.2 follows immediately from the second inequality in Theorem 3.3.1 together with the following lemma:

Lemma 3.5.2. *Fix an integer n and let D be a dyadic set in X_n. Then*

$$\rho(D) \geq 1.25^{-n-o(n)},$$

and thus

$$\rho_{\min}(n) \geq 1.25^{-n-o(n)}.$$

Proof. Fix a dyadic set D in X_n. Lemma 3.5.1 implies that there exists $0 < \beta < 1$ such that $\rho(D) \geq 2^{(2\beta - h(\beta))n - o(n)}$. Using Claim 3.4.2 we have $2^{2\beta - h(\beta)} \geq \frac{4}{5}$, and so

$$\rho(D) \geq (4/5)^n \cdot 2^{-o(n)} = 1.25^{-n-o(n)}.$$

$\square$

4 Positive prolixity

In the previous section we studied the minimum size of a set $\mathcal{Q}$ of questions with the property that for every distribution, there is an optimal decision tree using only questions from $\mathcal{Q}$. In this section we relax this requirement by allowing the cost to be slightly worse than the optimal cost.

More formally, recall that $u(n, r)$ is the minimum size of a set of questions $\mathcal{Q}$ such that for every distribution π there exists a decision tree that uses only questions from $\mathcal{Q}$ with cost at most $\mathrm{Opt}(\pi) + r$.

We focus here on the range $r \in (0, 1)$. We prove the following bounds on $u(n, r)$, establishing that $u(n, r) \approx (rn)^{\Theta(1/r)}$.

Theorem 4.1. *For all $r \in (0, 1)$, and for all $n > 1/r$:*

$$\frac{1}{n}(rn)^{\frac{1}{4r}} \leq u(n, r) \leq n^2 (3rn)^{\frac{16}{r}}.$$

As a corollary, we get that the threshold of exponentiality is $1/n$:

Corollary 4.2. *If $r = \omega(1/n)$ then $u(n, r) = 2^{o(n)}$.*
Conversely, if $r = O(1/n)$ then $u(n, r) = 2^{\Omega(n)}$.

For completeness, let us mention the following result, which follows from [7] via the trivial bound $u(n, r) \leq u_H(n, r) \leq u(n, r - 1)$, where $u_H(n, r)$ is the minimal size of a set of questions using which every distribution μ has a decision tree of cost at most $H(\mu) + r$:

Theorem 4.3. *For every $r \geq 1$ and all n,*

$$\frac{1}{e}\lfloor r+1 \rfloor n^{1/\lfloor r+1 \rfloor} \leq u(n, r) \leq 2\lfloor r \rfloor n^{1/\lfloor r \rfloor}.$$

Theorem 4.1 is implied by the following lower and upper bounds, which provide better bounds when $r \in (0, 1)$ is a negative power of 2.

Theorem 4.4 (Lower bound)**.** *For every r of the form $1/2^k$, where $k \geq 1$ is an integer, and $n > 2^k$:*

$$u(n,r) \geq (rn)^{\frac{1}{2r}-1}.$$

Theorem 4.5 (Upper bound)**.** *For every r of the form $4/2^k$, where $k \geq 3$ is an integer, and $n > 2^k$:*

$$u(n, r+r^2) \leq n^2\big(\frac{3e}{4}rn\big)^{\frac{4}{r}}.$$

These results imply Theorem 4.1, due to the monotonicity of $u(n,r)$, as follows.

Let $r \in (0,1)$. For the *lower bound*, pick the smallest $t \geq r$ of the form $1/2^k$. Note that $t \leq 2r$, and thus:

$$u(n,r) \geq u(n,t) \geq (tn)^{\frac{1}{2t}-1} \geq (rn)^{\frac{1}{4r}-1} \geq \frac{1}{n}(rn)^{\frac{1}{4r}}.$$

For the *upper bound*, pick the largest t of the form $4/2^k$, $k \geq 3$ such that $t + t^2 \leq r$. Note that $t \geq r/4$ (since $s = r/2$ satisfies $s + s^2 \leq 2s \leq r$), and thus

$$u(n,r) \leq u(n, t+t^2) \leq n^2\big(\frac{3e}{4}tn\big)^{\frac{4}{t}} \leq n^2(3rn)^{\frac{16}{r}}.$$

4.1 Lower bound

Pick a sufficiently small $\delta > 0$ (as we will soon see, $\delta < r^2$ suffices), and consider a distribution μ with $2^k - 1$ "heavy" elements (this many elements exist since $n > 1/r$), each of probability $\frac{1-\delta}{2^k-1}$, and $n-(2^k-1)$ "light" elements with total probability of δ. Recall that a decision tree is r-*optimal* if its cost is at most $\mathrm{Opt}(\mu)+r$. The proof proceeds by showing that if T is an r-optimal tree, then the first question in T has the following properties:

(i) it separates the heavy elements to two sets of almost equal sizes (2^{k-1} and $2^{k-1}-1$), and

(ii) it does not distinguish between the light elements.

The result then follows since there are $\binom{n}{2^k-1}$ such distributions σ (the number of ways to choose the light elements), and each question can serve as a first question to at most $\binom{n-(2^{k-1}-1)}{2^{k-1}}$ of them.

To establish these properties, we first prove a more general result (cf. Lemma 3.2.5):

Lemma 4.1.1. *Let μ be a distribution over a finite set X, and let $A \subseteq X$ be such that for every $x \notin A$, $\mu\big(\{x\}\big) > \mu(A) + \epsilon$. Then every decision tree T which is ϵ-optimal with respect to μ has a subtree T' whose set of leaves is A.*

Proof. By induction on $|A|$. The case $|A| = 1$ follows since any leaf is a subtree. Assume $|A| > 1$. Let T be a decision tree which is ϵ-optimal with respect to μ. Let x, y be two siblings of maximal depth. Note that it suffices to show that $x, y \in A$, since then, merging x, y to a new element z with $\mu(\{z\}) = \mu(\{x\}) + \mu(\{y\})$ and applying the induction hypothesis yields that $A \cup \{z\} \setminus \{x,y\}$ is the set of leaves of a subtree of T with x, y removed. This finishes the proof since x, y are the children of z.

It remains to show that $x, y \in A$. Let d denote the depth of x and y. Assume towards contradiction that $x \notin A$. Pick $a', a'' \in A$, with depths d', d'' (this is possible since $|A| > 1$). If $d' < d$ or $d'' < d$ then replacing a' with x or a'' with x improves the cost of T by more than ϵ, contradicting its optimality. Therefore, it must be that $d' = d'' = d$, and we perform the following transformation (see Figure 2): the parent of x and y becomes a leaf with label x (decreasing the depth of x by 1), y takes the place of a' (the depth of y does not change), and a'' becomes

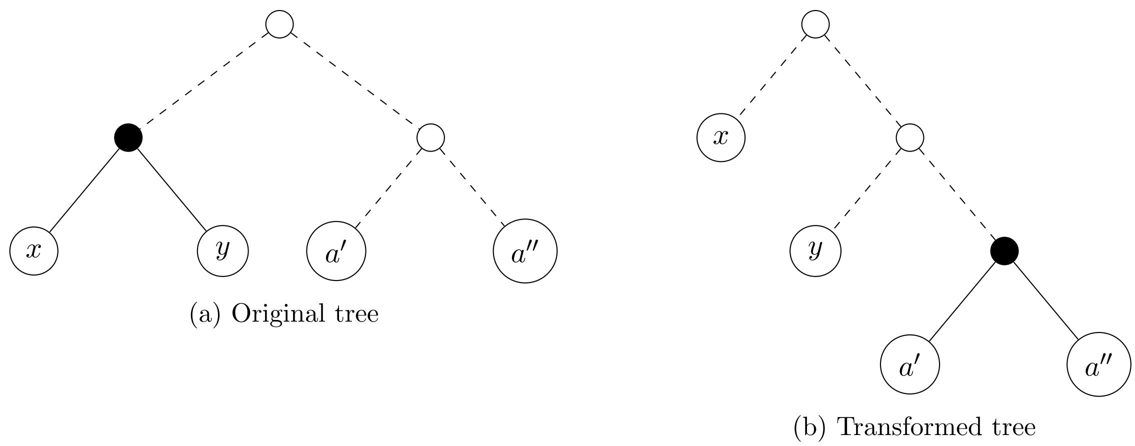

Figure 2: The transformation in Lemma 4.1.1. The cost decreases by $\mu(\{x\}) - \mu(\{a', a''\}) > \epsilon$.

an internal node with two children labeled by a', a'' (increasing the depths of a', a'' by 1). Since $\mu(\{x\}) - \mu(\{a', a''\}) > \epsilon$, this transformation improves the cost of T by more than ϵ, contradicting its ϵ-optimality. $\square$

Corollary 4.1.2. *Let μ be a distribution over X, and let $A \subseteq X$ be such that for every $x \notin A$, $\mu(\{x\}) > \mu(A)$. Then every optimal tree T with respect to μ has a subtree T' whose set of leaves is A.*

Property (ii) follows from Lemma 4.1.1, which implies that if δ is sufficiently small then all light elements are clustered together as the leaves of some subtree. Indeed, by Lemma 4.1.1, this happens if the probability of a single heavy element (which is $\frac{1-\delta}{2^k-1}$) exceeds the total probability of all light elements (which is δ) by at least r. A simple calculation shows that setting δ smaller than r^2 suffices.

We summarize this in the following claim:

Claim 4.1.3 (light elements)**.** *Every r-optimal tree has a subtree whose set of leaves is the set of light elements.*

The next claim concerns the other property:

Claim 4.1.4 (heavy elements)**.** *In every r-optimal decision tree, the first question partitions the heavy elements into a set of size 2^{k-1} and a set of size $2^{k-1} - 1$.*

Proof. When $k = 2$, it suffices to prove that an r-optimal decision tree cannot have a first question which separates the heavy elements from the light elements. Indeed, the heavy elements in such a tree reside at depths $2, 3, 3$. Exchanging one of the heavy elements at depth 2 with the subtree consisting of all light elements (which is at depth 1) decreases the cost by $\frac{1-\delta}{2^k-1} - \delta > r$, showing that the tree wasn't r-optimal.

Suppose that some r-optimal decision tree T contradicts the statement of the claim, for some $k \geq 3$. The first question in T leads to two subtrees T_1, T_2, one of which (say T_1) contains at least $2^{k-1} + 1$ heavy elements, while the other (say T_2) contains at most $2^{k-1} - 2$. One of the subtrees also contains a subtree T' whose leaves are all the light elements. For the sake of the argument, we replace the subtree T' with a new element y.

We claim that T_1 contains an internal node v at depth $D(v) \geq k - 1$ which has at least two heavy descendants. To see this, first remove y if it is present in T_1, by replacing its parent by its sibling. The possibly modified tree T_1' contains at least $2^{k-1}+1$ leaves, and in particular some leaf at depth at least k. Its parent v has depth at least $k-1$ and at least two heavy descendants, in both T_1' and T_1.

In contrast, T_2 contains at least two leaves (since $2^{k-1}-2 \geq 2$), and the two shallowest ones must have depth at most $k-2$. At least one of these is some heavy element x_ℓ.

Exchanging v and x_ℓ results in a tree T^* whose cost $c(T^*)$ is at most

$$c(T^*) \leq c(T) + (D(v) - D(x_\ell))(2-1)\frac{1-\delta}{2^k-1} \leq c(T^*) - \frac{1-\delta}{2^k-1} < c(T) - r,$$

contradicting the assumption that T is r-optimal. (That $\frac{1-\delta}{2^k-1} > r$ follows from the earlier assumption $\frac{1-\delta}{2^k-1} > \delta + r$.) □

By the above claims, there are two types of first questions for μ, depending on which of the two subtrees of the root contains the light elements:

- Type 1: questions that split the elements into a part with 2^{k-1} elements, and a part with $n - 2^{k-1}$ elements.
- Type 2: questions that split the elements into a part with $2^{k-1}-1$ elements, and a part with $n-(2^{k-1}-1)$ elements.

If we identify a question with its smaller part (i.e. the part of size 2^{k-1} or the part of size $2^{k-1}-1$), we deduce that any set of questions with redundancy r must contain a family $\mathcal{F}$ such that (i) every set in $\mathcal{F}$ has size 2^{k-1} or $2^{k-1}-1$, and (ii) for every set of size $n-(2^k-1)$, there exists some set in $\mathcal{F}$ that is disjoint from it. It remains to show that any such family $\mathcal{F}$ is large.

Indeed, there are $\binom{n}{2^k-1}$ sets of size $n-(2^k-1)$, and since every set in $\mathcal{F}$ has size at least $2^{k-1}-1$, it is disjoint from at most $\binom{n-(2^{k-1}-1)}{n-(2^k-1)} = \binom{n-(2^{k-1}-1)}{2^{k-1}}$ of them. Thus

$$|\mathcal{F}| \geq \frac{\binom{n}{2^k-1}}{\binom{n-(2^{k-1}-1)}{2^{k-1}}} = \frac{n(n-1)\cdots(n-(2^{k-1}-1)+1)}{(2^k-1)(2^k-2)\cdots(2^{k-1}+1)} \geq \left(\frac{n}{2^k}\right)^{2^{k-1}-1} = (rn)^{\frac{1}{2r}-1}.$$

4.2 Upper bound

The set of questions. In order to describe the set of queries it is convenient to assign a cyclic order on X_n: $x_1 \prec x_2 \prec \cdots \prec x_n \prec x_1 \prec \cdots$. The set of questions $\mathcal{Q}$ consists of all cyclic intervals, with up to 2^k elements added or removed. Since $r = 4 \cdot 2^{-k}$, the number of questions is plainly at most

$$n^2 \binom{n}{2^k} 3^{2^k} \leq n^2 \left(\frac{3e}{4} rn\right)^{\frac{4}{r}},$$

using the inequality $\binom{n}{d} \leq \left(\frac{en}{d}\right)^d$.

High level of the proof. Let π be an arbitrary distribution on X_n, and let $r \in (0,1)$ be of the form $4 \cdot 2^{-k}$, with $k \geq 3$. Let μ be a Huffman distribution for π; we remind the reader that μ is a dyadic distribution corresponding to some optimal decision tree for π. We construct a decision tree T that uses only queries from $\mathcal{Q}$, with cost

$$T(\pi) \leq \mathrm{Opt}(\pi) + r + r^2 = \sum_{x \in X_n} \pi(x) \log \frac{1}{\mu(x)} + r + r^2.$$

The construction is randomized: we describe a randomized decision tree T_R ('R' denotes the randomness that determines the tree) which uses queries from $\mathcal{Q}$ and has the property that for every $x \in X_n$, the expected number of queries T_R uses to find x satisfies the inequality

$$\mathop{\mathbb{E}}_{R}[T_R(x)] \leq \log \frac{1}{\mu(x)} + r + r^2, \tag{1}$$

where $T_R(x)$ is the depth of x. This implies the existence of a deterministic tree with cost $\mathrm{Opt}(\mu) + r + r^2$: indeed, when $x \sim \mu$, the expected cost of T_R is

$$\mathop{\mathbb{E}}_{x\sim\pi;R}[T_R(x)] \leq \sum_{x \in X_n} \pi(x)\Big(\frac{1}{\mu(x)} + r + r^2\Big) = \mathrm{Opt}(\pi) + r + r^2.$$

Since the randomness of the tree is independent from the randomness of π, it follows that there is a choice of R such that the cost of the (deterministic) decision tree T_R is at most $\mathrm{Opt}(\pi) + r + r^2$.

The randomized decision tree. The randomized decision tree maintains a dyadic *sub-distribution* $\mu^{(i)}$ that is being updated after each query. A *dyadic sub-distribution* is a measure on X_n such that (i) $\mu^{(i)}(x)$ is either 0 or a power of 2, and (ii) $\mu^{(i)}(X_n) = \sum_{x \in X_n} \mu^{(i)}(x) \leq 1$. A natural interpretation of $\mu^{(i)}(x)$ is as a dyadic sub-estimate of the probability that x is the secret element, conditioned on the answers to the first i queries. The analysis hinges on the following properties:

1. $\mu^{(0)} = \mu$,
2. $\mu^{(i)}(x) \in \big\{2\mu^{(i-1)}(x), \mu^{(i-1)}(x), 0\big\}$ for all $x \in X_n$,
3. if x is the secret element then almost always $\mu^{(i)}(x)$ is doubled; that is, $\mu^{(i)}(x) > 0$ for all i, and the expected number of i's for which $\mu^{(i)}(x) = \mu^{(i-1)}(x)$ is at most $r + r^2$.

These properties imply (1), which implies Theorem 4.5.

Next, we describe the randomized decision tree and establish these properties.

The algorithm distinguishes between light and heavy elements. An element $x \in X_n$ is *light* if $\mu^{(i)}(x) < 2^{-k}$. Otherwise it is *heavy*. The algorithm is based on the following win-win-win situation:

(i) If the total mass of the heavy elements is at least $1/2$ then by Lemma 2.1, there is a set I of heavy elements whose mass is exactly $1/2$. Since the number of heavy elements is at most 2^k, the algorithm can ask whether $x \in I$ and recurse by doubling the sub-probabilities of the elements that are consistent with the answer (and setting the others to zero).

(ii) Otherwise, the mass of the heavy elements is less than $1/2$. If the mass of the light elements is also less than $1/2$ (this could happen since $\mu^{(i)}$ is a sub-distribution), then we ask whether x is a heavy element or a light element, and accordingly recurse with either the heavy or the light elements, with their sub-probabilities doubled (in this case the "true" probabilities conditioned on the answers become larger than the sub-probabilities).

(iii) The final case is when the mass of the light elements is larger than $1/2$. In this case we query a random cyclic interval of light elements of mass $\approx 1/2$, and recurse; there are two light elements in the recursion whose sub-probability is not doubled (the probabilities of the rest are doubled).

Elements whose probability is not doubled occur only in case (iii).

The randomized decision tree: formal description. The algorithm gets as input a subset $y_1, \dots, y_m$ of X_n whose order is induced by that of X_n, and a dyadic sub-distribution $q_1, \dots, q_m$. Initially, the input is $x_1, \dots, x_n$, and $q_i = \mu_i$.

We say that an element is *heavy* is $q_i \geq 2^{-k}$; otherwise it is *light*. There are at most 2^k heavy elements. The questions asked by the algorithm are cyclic intervals in $y_1, \dots, y_m$, with some heavy elements added or removed. Since each cyclic interval in $y_1, \dots, y_m$ corresponds to a (not necessarily unique) cyclic interval in X_n (possibly including elements outside of $y_1, \dots, y_m$), these questions belong to $\mathcal{Q}$.

Algorithm T_R.

1. If $m = 1$, return y_1. Otherwise, continue to Step 2.
2. If the total mass of heavy elements is at least 1/2 then find (using Lemma 2.1) a subset I whose mass is exactly 1/2, and ask whether $x \in I$. Recurse with either $\{2q_i : y_i \in I\}$ or $\{2q_i : y_i \notin I\}$, according to the answer. Otherwise, continue to Step 3.
3. Let S be the set of all light elements, and let σ be their total mass. If $\sigma \leq 1/2$ then ask whether $x \in S$, and recurse with either $\{2q_i : y_i \in S\}$ or $\{2q_i : y_i \notin S\}$, according to the answer. Otherwise, continue to Step 4.
4. Arrange all light elements according to their cyclic order on a circle of circumference σ, by assigning each light element x_i an arc A_i of length q_i of the circle. Pick an arc of length 1/2 uniformly at random (e.g. by picking uniformly a point on the circle and taking an arc of length 1/2 directed clockwise from it), which we call the *window*. Let $K \subseteq S$ consist of all light elements whose midpoints are contained in the window, and let B consist of the light elements whose arcs are cut by the boundary of the window (so $|B| \leq 2$); we call these elements *boundary elements*. Ask whether $x \in K$; note that K is a cyclic interval in $y_1, \dots, y_m$ with some heavy elements removed.

   If $x \in K$, recurse with $\{2q_i : y_i \in K \setminus B\} \cup \{q_i : y_i \in K \cap B\}$. The sum of these dyadic probabilities is at most 1 since the window contains at least $q_i/2$ of the arc A_i for each $y_i \in K \cap B$.

   If $x \notin K$, recurse with $\{2q_i : y_i \in \overline{K} \setminus B\} \cup \{q_i : y_i \in \overline{K} \cap B\}$. As in the preceding case, the total mass of light elements in the recursion is at most $2(\sigma - 1/2)$ (since the complement of the window contains at least $q_i/2$ of the arc A_i for each $y_i \in \overline{K} \cap B$), and the total mass of heavy elements is $2(1 - \sigma)$, for a total of at most $(2\sigma - 1) + (2 - 2\sigma) = 1$. $\triangleleft$

Analysis. We now finish the proof by establishing the three properties of the randomized decision tree that are stated above. The first two properties follow immediately from the description of the algorithm, and it thus remains to establish the third property. Fix some $x \in X_n$, and let $d \in \mathbb{N}$ be such that $\mu(x) = 2^{-d}$. We need to show that the expected number of questions that are asked when the secret element is x is at most $d + r + r^2$.

Let $q = q^{(i)}$ denote the sub-probability of x after the i'th question; note that $q \in \{2^{-j} : j \leq d\}$.

Lemma 4.2.1. *If $q \geq 2^{-k}$ then q doubles (that is, $q^{(i+1)} = 2q^{(i)}$). Otherwise, the expected number of questions until q doubles is at most $\frac{1}{1-4q}$.*

Proof. From the description of the algorithm, it is clear that the only case in which the sub-probability of x is not doubled is when x is one of the two boundary elements in Step 4. This only happens when x is a light element (i.e. $q < 2^{-k}$). The probability that x is one of the boundary elements is at most $2q/\sigma \leq 4q$, where $\sigma \geq 1/2$ is the total mass of light elements: indeed, the probability that a given endpoint of the window lies inside the arc corresponding to q is q/σ, since each endpoint is distributed uniformly on the circle of circumference σ.

It follows that the distribution of the number of questions that pass until q doubles is dominated by the geometric distribution with failure probability $4q$, and so the expected number of questions until q doubles is at most $\frac{1}{1-4q}$. $\square$

The desired bound on the expected number of questions needed to find x follows from Lemma 4.2.1: as long as q, the sub-probability associated with x, is smaller than 2^{-k}, it takes an expected number of $\frac{1}{1-4q}$ questions until it doubles. Once $q \geq 2^{-k}$, it doubles after every question. Thus, by linearity of expectation, the expected total number of questions is at most:

$$\begin{aligned} k + \sum_{j=k+1}^{d} \frac{1}{1 - 4 \cdot 2^{-j}} &< k + \sum_{j=k+1}^{d} [1 + 4 \cdot 2^{-j} + 2(4 \cdot 2^{-j})^2] \\ &= d + \sum_{j=k+1}^{d} [4 \cdot 2^{-j} + 2(4 \cdot 2^{-j})^2] \\ &< d + 4 \cdot 2^{-k} + \frac{2}{3}(4 \cdot 2^{-k})^2 \\ &< \log \frac{1}{\mu(x)} + r + r^2. \end{aligned}$$

5 Open questions

Our work suggests many open questions, some of which are:

1. The main results of Section 3 show that when $n = 5 \cdot 2^m$, $u(n, 0) = 1.25^{n \pm o(n)}$. We conjecture that there exists a function $G \colon [1, 2] \to \mathbb{R}$ such that for $n = \alpha 2^m$, $u(n, 0) = G(\alpha)^{n \pm o(n)}$. Our results show that $1.232 \leq G(\alpha) \leq 1.25$ and that $G(1.25) = 1.25$. What is the function G?

2. Theorem 3.2 constructs an optimal set of questions of size $1.25^{n+o(n)}$, but this set is not explicit. In contrast, Theorem 3.5 constructs explicitly an optimal set of questions of size $O(\sqrt{2}^n)$, which furthermore supports efficient indexing and efficient construction of optimal strategies. Can we construct such an explicit set of optimal size $1.25^{n+o(n)}$?

3. Theorem 4.3 gives bounds on $u(n, r)$ for $r \geq 1$. For integer r, the lower bound is $\Omega(rn^{1/(r+1)})$, and the upper bound is $O(rn^{1/r})$. What is the correct exponent of n?

References

[1] Rudolf Ahlswede and Ingo Wegener. *Search problems.* John Wiley & Sons, Inc., New York, 1987.

[2] Javad A. Aslam and Aditi Dhagat. Searching in the presence of linearly bounded errors. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing (STOC '91)*, pages 486–493, 1991.

[3] Harout Aydinian, Ferdinando Cicalese, and Christian Deppe, editors. *Information Theory, Combinatorics, and Search Theory.* Springer-Verlag Berlin Heidelberg, 2013.

[4] David W. Boyd. The asymptotic number of solutions of a diophantine equation from coding theory. *Journal of Combinatorial Theory, Series A*, 18:210–215, 1975.

[5] Renato M. Capocelli, Raffaele Giancarlo, and Indeer Jeet Taneja. Bounds on the redundancy of Huffman codes. *IEEE Transactions on Information Theory*, IT-32(6):854–857, 1986.

[6] Thomas M. Cover and Joy A. Thomas. *Elements of information theory (2. ed.).* Wiley, 2006.

[7] Yuval Dagan, Yuval Filmus, and Shay Moran. Comparison and equality queries achieve optimal redundancy. In preparation.

[8] Aditi Dhagat, Peter Gács, and Peter Winkler. On playing "twenty questions" with a liar. In *Proceedings of 3rd Symposium on Discrete Algorithms (SODA'92)*, pages 16–22, 1992.

[9] Robert Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4):436–440, 1943.

[10] Ding-Zhu Du and Frank K. Hwang. *Combinatorial Group Testing and Its Applications*, volume 12 of *Series on Applied Mathematics.* World Scientific, 2nd edition, 1999.

[11] Michael L. Fredman. How good is the information theory bound in sorting? *Theoretical Computer Science*, 1(4):355–361, 1976.

[12] Robert G. Gallager. Variations on a theme by Huffman. *IEEE Transactions on Information Theory*, IT-24(6):668–674, 1978.

[13] E. N. Gilbert and E. F. Moore. Variable-length binary encodings. *Bell System Technical Journal*, 38:933–967, 1959.

[14] Yasuichi Horibe. An improved bound for weight-balanced tree. *Information and Control*, 34(2):148–151, 1977.

[15] David A. Huffman. A method for the construction of minimum-redundancy codes. In *Proceedings of the I.R.E.*, pages 1098–1103, 1952.

[16] Ottar Johnsen. On the redundancy of binary Huffman codes. *IEEE Transactions on Information Theory*, IT-26(2):220–222, 1980.

[17] Gyula O. H. Katona. Combinatorial search problems. In J. N. Srivastava et al., editor, *A Survery of Combinatorial Theory.* North-Holland Publishing Company, 1973.

[18] Philip Klein and Neal E. Young. On the number of iterations for Dantzig–Wolfe optimization and packing-covering approximation algorithms. *SIAM Journal on Computing*, 44(4):1154–1172, 2015.

[19] Daniel Krenn and Stephan Wagner. Compositions into powers of b: asymptotic enumeration and parameters. *Algorithmica*, 75(4):606–631, August 2016.

[20] Zbigniew Lonc and Ivan Rival. Chains, antichains, and fibres. *Journal of Combinatorial Theory, Series A*, 44:207–228, 1987.

[21] Dietrich Manstetten. Tight bounds on the redundancy of Huffman codes. *IEEE Transactions on Information Theory*, IT-38(1):144–151, 1992.

[22] Soheil Mohajer, Payam Pakzad, and Ali Kakhbod. Tight bounds on the redundancy of Huffman codes. In *Information Theory Workshop (ITW '06)*, pages 131–135, 2006.

[23] Bruce L. Montgomery and Julia Abrahams. On the redundancy of optimal binary prefix-condition codes for finite and infinite sources. *IEEE Transactions on Information Theory*, IT-33(1):156–160, 1987.

[24] Shay Moran and Amir Yehudayoff. A note on average-case sorting. *Order*, 33(1):23–28, 2016.

[25] Narao Nakatsu. Bounds on the redundancy of binary alphabetical codes. *IEEE Transactions on Information Theory*, IT-37(4):1225–1229, 1991.

[26] Jorma Rissanen. Bounds for weight balanced trees. *IBM Journal of Research and Development*, 17:101–105, 1973.

[27] Ronald L. Rivest, Albert R. Meyer, Daniel J. Kleitman, Karl Winklmann, and Joel Spencer. Coping with errors in binary search procedures. *Journal of Computer and System Sciences*, 20:396–404, 1980.

[28] Joel Spencer and Peter Winkler. Three thresholds for a liar. *Combinatorics, Probability and Computing*, 1(1):81–93, 1992.