SIAM J. COMPUT.
Vol. 42, No. 6, pp. 2182–2196

A SPACE LOWER BOUND FOR DYNAMIC APPROXIMATE MEMBERSHIP DATA STRUCTURES*

SHACHAR LOVETT† AND ELY PORAT‡

Abstract. An approximate membership data structure is a randomized data structure representing a set which supports membership queries. It allows for a small false positive error rate but has no false negative errors. Such data structures were first introduced by Bloom in the 1970s and have since had numerous applications, mainly in distributed systems, database systems, and networks. The algorithm of Bloom (known as a Bloom filter) is quite effective: it can store an approximation of a set S of size n by using only $\approx 1.44n\log_2(1/\varepsilon)$ bits while having false positive error ε. This is within a constant factor of the information-theoretic lower bound of $n\log_2(1/\varepsilon)$ for storing such sets. Closing this gap is an important open problem, as Bloom filters are widely used in situations where storage is at a premium. Bloom filters have another property: they are dynamic. That is, they support the iterative insertions of up to n elements. In fact, if one removes this requirement, there exist static data structures that receive the entire set at once and can almost achieve the information-theoretic lower bound; they require only $(1+o(1))n\log_2(1/\varepsilon)$ bits. Our main result is a new lower bound for the space requirements of any dynamic approximate membership data structure. We show that for any constant $\varepsilon > 0$, any such data structure that achieves false positive error rate of ε must use at least $C(\varepsilon)\cdot n\log_2(1/\varepsilon)$ memory bits, where $C(\varepsilon) > 1$ depends only on ε. This shows that the information-theoretic lower bound cannot be achieved by dynamic data structures for any constant error rate.

Key words. Bloom filters, dynamic data structures, lower bounds

AMS subject classifications. 68P05, 68Q17

DOI. 10.1137/120867044

1. Introduction.

Suppose we want to build a data structure that, given a set of elements $S = \{x_1, \ldots, x_n\}$ in a universe set U and an additional element $y \in U$, will be able to determine whether $y \in S$ or not. The *approximate membership problem* consists of storing a data structure that supports membership queries in the following manner: For a query on $y \in S$ it is always reported that $y \in S$. For a query on $y \notin S$ it is reported with probability at least $1-\varepsilon$ that $y \notin S$, and with probability at most ε that $y \in S$. That is, an approximate membership data structure has no false negative errors and allows false positive errors with probability at most ε. The probability is over the randomized choice of the data structure and not any random choices of the query algorithm. Hence, repeating a query may not lower the error probability.

The approximate membership problem has attracted significant interest in recent years, since it is a common building block for various applications, mainly in distributed systems, database systems, and networks (see [BM03] for a survey). Approximate membership data structures are often used in practice when storage is at a premium, while a small probability for false positive errors can be tolerated. The

*Received by the editors February 22, 2012; accepted for publication (in revised form) August 29, 2013; published electronically December 5, 2013. An extended abstract of this paper appeared in *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, IEEE Computer Society, Washington, DC, 2010, pp. 797–804.

http://www.siam.org/journals/sicomp/42-6/86704.html

†Computer Science and Engineering Department, University of California, San Diego, La Jolla, CA 92093 (slovett@cse.ucsd.edu). This author's research was supported by NSF grant DMS-0835373.

‡Computer Science Department, Bar-Ilan University, Ramat Gan, 52900, Israel (porately@cs.biu.ac.il). This author's research was supported by the Israel Science Foundation, the United States-Israel Binational Science Foundation, and a Google award.

false positive error rate that can be tolerated is often relatively large, say, in the range of 1%–10%.

The study of approximate membership was initiated by Bloom [Blo70], who described the *Bloom filter* data structure, which provides a simple and elegant solution for the problem that is near-optimal. Bloom showed that a space usage[1] of $n \log_2(1/\varepsilon) \log_2 e$ bits suffices for a false positive error probability of ε (for ε an integer power of two). This is quite close to the information-theoretic lower bound. Carter et al. [CFG+78] showed that $n \log_2(1/\varepsilon)$ bits are required when the universe set $\mathbf{U}$ is large, e.g., $n = o(|\mathbf{U}|)$ (see also [DP08] for details). Thus Bloom filters have a space usage within a factor of $\log_2 e \approx 1.44$ of the lower bound. As Bloom filters are widely used in practice, mainly in situations when storage is scarce, this factor of 1.44 is not negligible. The main object of study in this paper is whether this factor can be eliminated; i.e., we study whether there exist data structures for approximate membership that achieve the information-theoretic lower bound.

An important feature of Bloom filters is that they are *dynamic*. That is, the elements $x_1, \ldots, x_n$ can be inserted one at a time, while maintaining the succinct representation of the data structure. If, on the other hand, we limit ourselves to *static* data structures, which are given the entire set $S = \{x_1, \ldots, x_n\}$ at once, and are allowed to preprocess it before creating the succinct data structure, then the information-theoretic lower bound can be nearly achieved. Dietzfelbinger and Pagh [DP08] and Porat [Por09] gave data structures for the static approximate membership problem using only $(1 + o(1))n \log_2(1/\varepsilon)$ bits.

The main result of this paper is that dynamic data structures for approximate membership *cannot* achieve the information-theoretic lower bound.

THEOREM 1.1. *Let* $\mathbf{U}$ *be a universe set. Consider any randomized data structure which allows for the dynamic insertion of up to* n *elements (where* $n = o(|\mathbf{U}|)$*), has false positive error at most* ε *(where* $\varepsilon > 0$ *is a constant), and which allows no false negative errors. Then for large enough* n*, any such data structure must use at least* $C(\varepsilon)n \log_2(1/\varepsilon)$ *memory bits, where* $C(\varepsilon) > 1$ *is a constant depending only on* ε*. In particular,*

- $C(1/2) \geq 1.1$.
- *For all sufficiently small* $\varepsilon > 0$, $C(\varepsilon) \geq 1 + \Omega(\frac{1}{\log^2(1/\varepsilon)})$.

We note that the requirement that the false negative error be constant cannot be eliminated. In fact, for every $\varepsilon = o(1)$ there is a simple dynamic approximate membership data structure that requires only $(1 + o(1))n \log_2(1/\varepsilon)$ bits: pick a (sufficiently good) hash function $h : \mathbf{U} \to [n/\varepsilon]$, and at the ith step maintain the set $\{h(x_1), h(x_2), \ldots, h(x_i)\}$. The space requirements of this algorithm are $\log_2 \binom{n/\varepsilon}{n} = n \log_2(1/\varepsilon) + O(n)$, which is $(1 + o(1))n \log_2(1/\varepsilon)$ for any $\varepsilon = o(1)$. The data structure we just described is not efficient; efficient versions are achieved implicitly in the work of Matias and Porat [MP07], explicitly in the work of Pagh, Pagh, and Rao [PPR05] (which is based on a work of Raman and Rao [RR03]), and in the work of Arbitman, Naor, and Segev [ANS10].

1.1. Proof overview. The proof of the lower bound is conducted in two steps. First, we transform the problem into a graph-theoretic problem, and then we prove results on this graph-theoretic problem.

[1] Actually, Bloom's solution assumes random hash functions. For concrete solutions that nearly meet this bound, see [MV08].

The graph-theoretic problem. Assume there exists a dynamic approximate membership data structure, which allows insertion of up to n elements from a universe set $\mathbf{U}$, has false positive error of at most ε, and which requires M memory bits. In order to prove a lower bound, it suffices to find a distribution on the inputs which is hard for any deterministic data structure. Thus, it suffices to study deterministic data structures. We model such a data structure by a labeled layered graph that captures all possible insertions of up to n elements. It is essentially the state graph of the finite automaton derived from the reachable state transitions of the data structure.

The graph G has $n+1$ layers $V_0 \cup V_1 \cup \cdots \cup V_n$, where each vertex in V_i corresponds to a possible state of the data structure after insertions of i elements. In particular, $V_0 = \{v_0\}$ and $|V_1|, \ldots, |V_n| \leq 2^M$. The edges connect vertices in adjacent layers and are labeled by elements $x \in \mathbf{U}$. Given a vertex $v \in V_i$ and an element $x \in \mathbf{U}$, there is an outgoing edge $v \to u$ that is labeled with x, where $u \in V_{i+1}$ corresponds to the state reached after inserting x when the state of the data structure was v. Thus, any sequence $w = x_1, \ldots, x_i \in \mathbf{U}^i$ defines a path from $v_0 \in V_0$ to some vertex $v(w) \in V_i$.

We now state an important definition. For a vertex v define $L(v) \subset \mathbf{U}$ to be the set of all labels in all paths between v_0 and v. In particular, for a sequence $w = x_1, \ldots, x_n$, the set $L(v(w))$ is the set of all labels that belong to sequences that also reach the same final state $v(w)$. We claim that the assumptions that the data structure has no false negative errors, and has false positive errors with probability at most ε, guarantee that $|L(v(w))| \leq (\varepsilon + o(1))|\mathbf{U}|$. The proof is simple: if $x \in L(v(w))$, then by definition, x belongs to some sequence $\widetilde{w} = \widetilde{x}_1, \ldots, \widetilde{x}_n$ for which $v(\widetilde{w}) = v(w)$. Thus, at the final state $v(w)$ we must report that $x \in S$, since the data structure has no false negatives. As the data structure has false positive error of ε, there could be at most $\varepsilon|\mathbf{U}| + n$ such elements. Thus we have $|L(v(w))| \leq (\varepsilon + o(1))|\mathbf{U}|$.

In the case of *randomized* data structures, we prove by an averaging argument that for some fixing of the internal randomness of the data structure, the *average* size of $|L(v(w))|$ is bounded,

$$\mathbb{E}_{w \in \mathbf{U}^n}[|L(v(w))|] \leq (\varepsilon + o(1))|\mathbf{U}|.$$

In fact, our lower bound holds even when one assumes access to random hash functions (as in the original work of Bloom [Blo70]).

Lower bound on the layers sizes. Let G be the labeled layered graph we constructed. We would like to prove that any such graph must have at least one layer with many vertices. This corresponds to the data structure requiring many memory bits at some step in the insertion process. Note that it is not enough to prove that the last layer must be large without restricting the size of other layers; this corresponds to proving a lower bound on static data structures.

Fix $1 \leq k \leq n$ to be some intermediate layer to be chosen later. We will show that at least one of either the kth layer or the last layer must be large. That is,

$$\max(|V_k|, |V_n|) \geq (1/\varepsilon)^{C(\varepsilon)n}.$$

Let $\max(|V_k|, |V_n|) = 2^M$, where we will lower bound M. Pick $w = x_1, \ldots, x_n \in \mathbf{U}^n$ uniformly at random. Note that this sequence of insertion might have duplicate entries. Of course, with large enough universe set the probability for duplicates is negligible.

Partition w to the first k elements $w' = x_1, \ldots, x_k$ and the last $n - k$ elements $w'' = x_{k+1}, \ldots, x_n$. We consider inserting the elements in w as a two-step process: first insert w', reaching an intermediate vertex $v(w') \in V_k$, and then insert w'', reaching a

final vertex $v(w'w'') \in V_n$. Recall that by assumption, $\mathbb{E}[L(v(w'w''))] \leq (\varepsilon + o(1))|\mathbf{U}|$. Thus, with not too small probability (say, at least $1/n$) over w, we have

$$|L(v(w'w''))| < \alpha|\mathbf{U}|, \tag{1}$$

where $\alpha = \varepsilon + o(1)$.

We next claim that $L(v(w'))$ cannot be too small with any significant probability. Let $v \in V_k$ and let $|L(v)| = \beta(v)|\mathbf{U}|$. The number of $w' \in \mathbf{U}^k$ for which $v(w') = v$ can be at most $\beta(v)^k|\mathbf{U}|^k$, since any element in w' must belong to $L(v)$. As any $w' \in \mathbf{U}^k$ reaches some $v(w') \in V_k$, we must have $\sum_{v \in V_k} \beta(v)^k \geq 1$. We assume that the number of vertices in V_k is at most 2^M. Thus, we cannot have $\beta(v) \ll 2^{-M/k}$ for many vertices $v(w')$. Formally, we show that with very good probability over w' we have

$$|L(v(w'))| \geq \beta|\mathbf{U}|, \tag{2}$$

where $\beta = 2^{-M/k} - o(1)$.

We will prove the lower bound by a covering argument, based on the above properties. Essentially, we will count possible paths restricted to layers k and n. We first sketch a simple covering argument, which fails at giving a lower bound better than the information-theoretic lower bound. We then show a more complex covering argument which will give a nontrivial lower bound on M.

We first consider the simple covering argument. Fix some $v' = v(w')$ and $v'' \in V_n$. If w'' is such that $v'' = v(w'w'')$, then we must have all elements in w'' appear in $L(v'')$. However, since $|L(v'')| \approx \varepsilon|\mathbf{U}|$, the number of possibilities for w'' is at most $(\varepsilon|\mathbf{U}|)^{n-k}$. Thus, since the total number of $w'' \in \mathbf{U}^{n-k}$ is $|\mathbf{U}|^{n-k}$, there must be at least $(1/\varepsilon)^{n-k}$ different vertices in V_n that can be reached from v'. This yields the bound $|V_n| \geq (1/\varepsilon)^{n-k}$, which is optimized by taking $k = 0$ and gives $M \geq n \log_2(1/\varepsilon)$. Note that this is the simple information-theoretic lower bound.

We now show how to obtain an improved covering argument. Say a sequence $w'' \in \mathbf{U}^{n-k}$ is *typical* for w' if w'' intersects $L(v(w'))$ in about "the right number" of times, that is, if

$$|w'' \cap L(v(w'))| \approx (n-k)\frac{L(v(w'))}{|\mathbf{U}|}. \tag{3}$$

Note that a simple Chernoff argument gives that most w'' are typical for w' (at least if $L(v(w'))/|\mathbf{U}|$ is not too close to 0, which we know by (2)).

Assume w'' is typical for w' such that $v'' = v(w'w'')$. Let $\beta(w') = \frac{L(v(w'))}{|\mathbf{U}|}$. The number of such w'' is bounded by

$$\approx \binom{n-k}{\beta(w')(n-k)} |L(v(w'))|^{\beta(w')(n-k)} |L(v'') \setminus L(v(w'))|^{(1-\beta(w'))(n-k)}.$$

We show that events (1), (2), and (3) all occur simultaneously with a relatively large probability. It can be shown that for a large fraction of w', there must be many distinct $v(w'w'')$ where w'' is typical for w',

$$|\{v(w'w'') : w'' \text{ is typical for } w'\}| \geq \left(\frac{1-\beta(w')}{\alpha - \beta(w')}\right)^{(1-\beta(w'))(n-k)}.$$

Using a convexity argument and the assumption that $\beta(w') \geq \beta$, we get that

$$|\{v(w'w'') : w'' \text{ is typical for } w'\}| \geq \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)}.$$

We next combine this with the simple bound that the number of $w' \in \mathbf{U}^k$ that can reach some vertex in a path to $v'' \in V_n$ is bounded by $|L(v'')|^k \leq (\alpha|\mathbf{U}|)^k$, and we deduce the following inequality. Set $c = \frac{k}{n}$ and $C = \frac{M}{n \log_2(1/\varepsilon)}$. Recall that our objective is to lower bound $C \geq C(\varepsilon)$ for some $C(\varepsilon) > 1$. We get

$$\left(\frac{1}{\varepsilon}\right)^C \varepsilon^c \geq \left(\frac{1-\varepsilon^{C/c}}{\varepsilon-\varepsilon^{C/c}}\right)^{(1-\varepsilon^{C/c})(1-c)}. \tag{4}$$

This is a nontrivial inequality relating the different parameters ε, c, C. Note that it should hold for any value of $0 < c < 1$. In the final step, we study inequality (4) and prove that for every constant $\varepsilon > 0$ we can choose some value for c such that we must have $C(\varepsilon) > 1$ for the inequality to hold. For specific values of ε one can optimize (4) by a computer search; for example, for $\varepsilon = 1/2$ we get that $C(\varepsilon) \geq 1.1$.

Paper organization. We formally define approximate membership data structures in section 2. We prove Theorem 1.1 in section 3. We summarize and state some open problems in section 4. In the appendix, we prove some technical analytical claims.

2. Preliminaries. Let $\mathbf{U}$ be a universe set. An *approximate membership* data structure is a randomized data structure that represents a subset $S \subset \mathbf{U}$ of size $|S| \leq n$ and supports queries of whether or not $x \in S$ for elements $x \in \mathbf{U}$, with the following guarantees:

- No false negatives: if $x \in S$, the query will always return *true*.
- Few false positives: if $x \notin S$, the query will return *false*, with probability at least $1-\varepsilon$, and will return *true*, with probability at most ε (probabilities are over the internal randomness of the data structure).

The main goal of this paper is to study the trade-off between the maximal set size n, the false positive error parameter ε, and the space requirements of the data structure. We will assume throughout the paper that the subset S is a small fraction of the universe, i.e., that $n = o(|\mathbf{U}|)$.

We now define *dynamic* versus *static* approximate membership data structures.

DEFINITION 2.1 (dynamic approximate membership data structure). *A dynamic approximate membership data structure is composed of two algorithms: an insertion algorithm and a query algorithm.*

- *The insertion algorithm $\mathcal{I}$ is a randomized algorithm, which allows for the insertion of up to n elements sequentially. The algorithm maintains a succinct representation R of the set of elements inserted so far, and for each new element $x \in \mathbf{U}$ updates $R \leftarrow \mathcal{I}(R, x)$.*
- *The query algorithm $\mathcal{Q}$ receives as inputs the succinct representation R of a set S and an element $x \in \mathbf{U}$, and outputs an estimate $\mathcal{Q}(R, x) \in \{true, false\}$ whether $x \in S$.*

The space requirements of a dynamic approximate membership data structure is the maximal number of bits required to represent R throughout the insertion phase. We denote by $M_D(n, \varepsilon)$ the minimal space required by a dynamic approximate membership data structure that stores up to n elements and has false positive errors with probability at most ε.

DEFINITION 2.2 (static approximate membership data structure). *A static approximate membership data structure is composed of two algorithms: a preprocessing algorithm and a query algorithm.*

- *The preprocessing algorithm $\mathcal{P}$ is a randomized algorithm, which receives as input a subset $S \subset \mathbf{U}$ of size at most n, and outputs a succinct representation $R = \mathcal{P}(S)$ of S.*
- *The query algorithm $\mathcal{Q}$ receives as inputs the succinct representation R of a set S and an element $x \in \mathbf{U}$, and outputs an estimate $\mathcal{Q}(R, x) \in \{true, false\}$ whether $x \in S$.*

The space requirements of a static approximate membership data structure is the number of bits required to represent $P(S)$. We denote by $M_S(n, \varepsilon)$ the minimal space required by a static approximate membership data structure that stores up to n elements and has false positive error with probability at most ε.

For the convenience of the reader we recap the known properties of the space requirements of dynamic and static approximate membership data structures. These include the information-theoretic lower bound of Carter et al. [CFG+78], Bloom filters [Blo70], and efficient static data structures of Dietzfelbinger and Pagh [DP08] and of Porat [Por09].

FACT 2.3. *For any constant $\varepsilon > 0$ we have*

- $M_S(n, \varepsilon) = (1 + o(1)) \cdot n \log_2(1/\varepsilon)$.
- $(1 - o(1)) \cdot n \log_2(1/\varepsilon) \leq M_D(n, \varepsilon) \leq \log_2 e \cdot n \log_2(1/\varepsilon) \approx 1.44 \cdot n \log_2(1/\varepsilon)$.

Our main result is an improved lower bound on $M_D(n, \varepsilon)$,

$$M_D(n, \varepsilon) \geq C(\varepsilon) \cdot n \log_2(1/\varepsilon),$$

where $C(\varepsilon) > 1$ is a constant depending only on ε.

3. Proof of Theorem 1.1.

3.1. The graph-theoretic problem. Let $(\mathcal{I}, \mathcal{Q})$ be the insertion and query randomized algorithms in an optimal dynamic approximate membership data structure for sets of size n with false positive error of ε, which uses $M = M_D(n, \varepsilon)$ memory bits. Let r denote the internal randomness used by the algorithms. We denote by $\mathcal{I}^r, \mathcal{Q}^r$ the algorithms given an explicit value r for the internal randomness.[2]

It will be convenient for us to model a dynamic approximate membership data structure by a labeled layered graph. For any fixing of the internal randomness r, define a labeled layered graph G^r as follows. The graph will have $n+1$ layers $V_0 \cup V_1 \cup \cdots \cup V_n$. Each vertex in V_i corresponds to a possible state of the data structure after insertions of i elements. In particular, $|V_0| = 1$ and $|V_1|, \ldots, |V_n| \leq 2^M$. The edges connect vertices in adjacent layers and are labeled by elements $x \in \mathbf{U}$. Given a vertex $v \in V_i$ and an element $x \in \mathbf{U}$, there is an outgoing edge $v \to u$ that is labeled with x, where $u = \mathcal{I}^r(v, x)$. Thus, the graph G^r describes all possible iterative insertions of n elements (given the fixing r of the internal randomness), and the collection of graphs $\{G^r\}$ is a complete description of the insertion algorithm.

For ease of notation, we extend the definition of $\mathcal{I}^r$ for sequences of elements. Let $w = x_1, \ldots, x_i \in \mathbf{U}^i$ be a sequence of i elements, and let $v \in V_j$ where $i + j \leq n$. We define $\mathcal{I}^r(v, w) \in V_{i+j}$ to be the vertex reached from v after insertion of $x_1, \ldots, x_i$, i.e.,

$$\mathcal{I}^r(v, w) = \mathcal{I}^r(\ldots \mathcal{I}^r(\mathcal{I}^r(v, x_1), x_2) \ldots, x_i).$$

[2]Our lower bounds hold even if one allows access to common random hash functions for the insertion and query algorithms.

We also shorthand $\mathcal{I}^r(w) = \mathcal{I}^r(v_0, w)$, where $v_0 \in V_0$ is the initial state of the data structure.

For a sequence $w = x_1, \ldots, x_n \in \mathbf{U}^n$, denote by $A^r(w)$ the set of all elements $x \in \mathbf{U}$ which are accepted by $\mathcal{Q}^r$ given the succinct representation $v = \mathcal{I}^r(w)$, i.e.,

$$A^r(w) = \{x \in \mathbf{U} : \mathcal{Q}^r(\mathcal{I}^r(w), x) = true\}.$$

The following claim summarizes the assumption that $(\mathcal{I}, \mathcal{Q})$ is a dynamic approximate membership data structure with no false negative errors and false positive errors with probability at most ε.

CLAIM 3.1. *Let $w = x_1, \ldots, x_n \in \mathbf{U}^n$. Then*

(1) *for any setting of r, we have $\{x_1, \ldots, x_n\} \subset A^r(w)$.*

(2) *fix $y \notin \{x_1, \ldots, x_n\}$. Then $\Pr_r[y \in A^r(w)] \le \varepsilon$.*

Proof. The first claim follows from the assumption that $(\mathcal{I}, \mathcal{Q})$ has no false negative errors. Thus, for any x_i $(i = 1, \ldots, n)$ since $x_i \in \{x_1, \ldots, x_n\}$ we must have that $\Pr_r[\mathcal{Q}^r(\mathcal{I}^r(w), x_i) = true] = 1$. The second claim follows from the assumption that $(\mathcal{I}, \mathcal{Q})$ have false positive errors with probability at most ε. Thus, for a random choice of r, $\Pr_r[\mathcal{Q}^r(\mathcal{I}^r(w), y) = true] \le \varepsilon$. ∎

As a corollary we get that the size of $A^r(w)$ must be small for average r.

CLAIM 3.2. *Let $w = x_1, \ldots, x_n \in \mathbf{U}^n$. Then $\mathbb{E}_r[|A^r(w)|] \le \varepsilon|\mathbf{U}| + n$.*

Proof. The proof follows immediately from Claim 3.1. Let $S = \{x_1, \ldots, x_n\}$. Then

$$\mathbb{E}_r[|A^r(w)|] = \sum_{y \in \mathbf{U}} \Pr_r[y \in A^r(w)] \le |S| + \sum_{y \in \mathbf{U} \setminus S} \Pr_r[y \in A^r(w)] \le n + \varepsilon|\mathbf{U}|. \quad ∎$$

We now fix the randomness for the algorithms. Let $w = x_1, \ldots, x_n \in \mathbf{U}^n$ be uniformly chosen. By Claim 3.2 we have in particular that

$$\mathbb{E}_r \mathbb{E}_{w \in \mathbf{U}^n}[|A^r(w)|] \le \varepsilon|\mathbf{U}| + n.$$

Thus, there must exist some fixing $r = r^*$ such that

$$\mathbb{E}_{w \in \mathbf{U}^n}[|A^{r^*}(w)|] \le \varepsilon|\mathbf{U}| + n.$$

From now on we fix the internal randomness to r^*, and for ease of notation omit the superscript r^* from $G, A, \mathcal{I}, \mathcal{Q}$. Hence we have the following corollary.

COROLLARY 3.3. $\mathbb{E}_{w \in \mathbf{U}^n}[|A(w)|] \le \varepsilon|\mathbf{U}| + n$.

3.2. Properties of the graph. We will prove some properties of the layered graph we obtained. These properties will later be used to prove the lower bound.

Let $0 < \delta \ll 1$ be a small parameter to be determined later. We first show that for a relatively large fraction of $w \in \mathbf{U}^n$, the set $A(w)$ is not much larger than the average size of these sets.

CLAIM 3.4. *Let $w \in \mathbf{U}^n$ be chosen uniformly. Set $\alpha = \varepsilon(1 + \frac{n}{|\mathbf{U}|})(1 + 6\delta) = (1 + o(1))\varepsilon$. Then*

$$\Pr_{w \in \mathbf{U}^n}[|A(w)| \le \alpha|\mathbf{U}|] \ge 3\delta.$$

Proof. Assume $\delta < 1/6$. Applying Markov's inequality,

$$\Pr_{w \in \mathbf{U}^n}[|A(w)| \ge \alpha|\mathbf{U}|] \le \frac{\mathbb{E}_{w \in \mathbf{U}^n}[|A(w)|]}{\alpha|\mathbf{U}|} = \frac{1}{1 + 6\delta} \le 1 - 3\delta. \quad ∎$$

We now state an important definition. Let $w = x_1, \ldots, x_i \in \mathbf{U}^i$ and let $v = \mathcal{I}(w)$. We define $L(w)$ to be the set of labels on any path that reaches v. That is,

$$L(w) = \{y \in \mathbf{U} : \exists \widetilde{w} = \widetilde{x}_1, \ldots, \widetilde{x}_i \in \mathbf{U}^i \text{ such that } \mathcal{I}(\widetilde{w}) = \mathcal{I}(w) \text{ and } y \in \{\widetilde{x}_1, \ldots, \widetilde{x}_i\}\}.$$

We now prove two useful properties of the sets $L(w)$.

CLAIM 3.5.

1. *Let $w = x_1, \ldots, x_i \in \mathbf{U}^i$ and $w' = x_{i+1}, \ldots, x_j \in \mathbf{U}^{i-j}$ for $i < j$. Let $ww' \in \mathbf{U}^j$ be the concatenation of w and w'. Then $L(w) \subseteq L(ww')$.*
2. *Let $w = x_1, \ldots, x_n \in \mathbf{U}^n$. Then $L(w) \subseteq A(w)$.*

Proof. The first property follows immediately from the definition of L. If $y \in L(w)$, then there exists $\widetilde{w} = \widetilde{x}_1, \ldots, \widetilde{x}_i \in \mathbf{U}^i$ such that $\mathcal{I}(\widetilde{w}) = \mathcal{I}(w)$ and $y \in \{\widetilde{x}_1, \ldots, \widetilde{x}_i\}$. But then $\mathcal{I}(\widetilde{w}w') = \mathcal{I}(ww')$; hence also $y \in L(ww')$.

The second property follows since a dynamic approximate membership data structure has no false negative errors. Let $y \in L(w)$, and let $\widetilde{w} = \widetilde{x}_1, \ldots, \widetilde{x}_n \in \mathbf{U}^n$ such that $\mathcal{I}(\widetilde{w}) = \mathcal{I}(w)$ and $y \in \{\widetilde{x}_1, \ldots, \widetilde{x}_n\}$. By Claim 3.1 we know that $\{\widetilde{x}_1, \ldots, \widetilde{x}_n\} \subset A(w)$. Hence also $y \in A(w)$. ∎

Let $1 \le k \le n$ be a parameter to be fixed later. We show that most sets $L(w)$ for $w \in \mathbf{U}^k$ cannot be too small.

CLAIM 3.6. *Let $w = x_1, \ldots, x_k \in \mathbf{U}^k$ be chosen uniformly. Then*

$$\Pr_{w \in \mathbf{U}^k}[|L(w)| \le \beta|\mathbf{U}|] \le \delta,$$

where $\beta = \delta^{1/k} 2^{-M/k}$.

Proof. The proof is by a simple counting argument. Let $\mathcal{L} = \{L(w) : w \in \mathbf{U}^k,\ |L(w)| \le \beta|\mathbf{U}|\}$ be the set of all possible $L(w)$ of size at most $\beta|\mathbf{U}|$. The size of $\mathcal{L}$ is at most 2^M as distinct sets in $\mathcal{L}$ match distinct vertices in V_k. For any set $\widetilde{L} \in \mathcal{L}$, we can have $L(w) = \widetilde{L}$ for $w = x_1, \ldots, x_k \in \mathbf{U}^k$ only if $\{x_1, \ldots, x_k\} \subset \widetilde{L}$. Thus, for any fixed $\widetilde{L}$, the number of such sequences is bounded by $(\beta|\mathbf{U}|)^k$. Hence,

$$\Pr_{w \in \mathbf{U}^k}[|L(w)| \le \beta|\mathbf{U}|] \le \frac{(\beta|\mathbf{U}|)^k 2^M}{|\mathbf{U}|^k} \le \delta. \qquad ∎$$

Let $w' = x_1, \ldots, x_k \in \mathbf{U}^k$ and $w'' = x_{k+1}, \ldots, x_n \in \mathbf{U}^{n-k}$. We denote by $C(w', w'')$ the number of elements in w'' which are in $L(w')$, i.e.,

$$C(w', w'') = |\{i : k+1 \le i \le n,\ x_i \in L(w')\}|.$$

We say w'' is *typical* for w' if $C(w', w'')$ is close to its expected size $\frac{|L(w')|}{|\mathbf{U}|}(n-k)$. The next claim shows that almost all w'' are typical for w'.

CLAIM 3.7. *Fix $w' = x_1, \ldots, x_k \in \mathbf{U}^k$. Let $w'' = x_{k+1}, \ldots, x_n \in \mathbf{U}^{n-k}$ be distributed uniformly at random. Then*

$$\Pr_{w'' \in \mathbf{U}^{n-k}}\left[\left|C(w', w'') - \frac{|L(w')|}{|\mathbf{U}|}(n-k)\right| \ge \gamma(n-k)\right] \le \delta,$$

where $\gamma = \sqrt{3\ln(2/\delta)/(n-k)}$.

In order to prove Claim 3.7 we will apply the Chernoff–Hoeffding bound, which we recall below.

LEMMA 3.8 (Chernoff–Hoeffding bound). *Let $X_1, \dots, X_m \in \{0,1\}$ be independent random variables such that $\mathbb{E}[X_i] = p$. Then for any $\gamma > 0$*

$$\Pr\left[\left|\frac{1}{m}\sum X_i - p\right| \geq \gamma\right] \leq 2e^{-\frac{\gamma^2}{3}m}.$$

Proof of Claim 3.7. Set $m = n-k$ and define X_i to be an indicator variable for $x_{k+i} \in L(w')$ for $i = 1, \dots, n-k$. Then $C(w', w'') = \sum_{i=1}^{n-k} X_i$, $\mathbb{E}_{w''}[X_i] = \frac{|L(w')|}{|\mathbf{U}|}$, and the Chernoff–Hoeffding bound gives

$$\Pr_{w'' \in \mathbf{U}^{n-k}}\left[\left|C(w', w'') - \frac{|L(w')|}{|\mathbf{U}|}(n-k)\right| \geq \gamma(n-k)\right] \leq 2e^{-\frac{\gamma^2}{3}(n-k)} \leq \delta. \qquad \square$$

We conclude Claims 3.4, 3.6, and 3.7 by the following claim, showing that there is a relatively large subset $W \subset \mathbf{U}^n$ for which all three claims hold simultaneously.

DEFINITION 3.9. *Let α, β, γ be as defined in Claims* 3.4, 3.6, *and* 3.7. *We define a subset of sequences $W \subset \mathbf{U}^n$ as follows. For $w \in \mathbf{U}^n$ write $w = w'w''$, where $w' \in \mathbf{U}^k$ and $w'' \in \mathbf{U}^{n-k}$. An element $w'w'' \in \mathbf{U}^n$ is in W if all of the following conditions hold:*

(i) $|L(w'w'')| \leq \alpha|\mathbf{U}|$.
(ii) $|L(w')| \geq \beta|\mathbf{U}|$.
(iii) $|C(w', w'') - \frac{|L(w')|}{|\mathbf{U}|}(n-k)| \leq \gamma(n-k)$.

CLAIM 3.10. $|W| \geq \delta|\mathbf{U}|^n$.

Proof. The proof is an immediate corollary of Claims 3.4, 3.5, 3.6, and 3.7. For uniformly chosen $w \in \mathbf{U}^n$, condition (i) holds with probability at least 3δ, and conditions (ii) and (iii) hold with probability at least $1-\delta$. Hence by the union bound all three hold simultaneously with probability at least δ. Hence $|W| \geq \delta|\mathbf{U}|^n$. $\square$

3.3. Inequalities on paths in the graph. We will prove a certain family on inequalities on the graph which relate to paths in the graph. Define X to be the set

$$X = \{(w', L(w'w'')) : w'w'' \in W\}.$$

We will prove lower and upper bounds on $|X|$ that will imply lower bounds on the space requirement M. We start with a simple upper bound.

CLAIM 3.11. $|X| \leq (\alpha|\mathbf{U}|)^k 2^M$.

Proof. The number of distinct sets $\{L(w) : w \in W\}$ is bounded by the size of the last level $|V_n| \leq 2^M$. Fix any $\widetilde{L} \in \{L(w) : w \in W\}$. By condition (i) we must have $|\widetilde{L}| \leq \alpha|\mathbf{U}|$. If $w' = x_1, \dots, x_k \in \mathbf{U}^k$ is such that $L(w'w'') = \widetilde{L}$, it must be that $\{x_1, \dots, x_k\} \subset \widetilde{L}$. Hence the number of such w' is at most $|\widetilde{L}|^k \leq (\alpha|\mathbf{U}|)^k$. Hence we conclude that $|X| \leq (\alpha|\mathbf{U}|)^k 2^M$. $\square$

For $w' \in \mathbf{U}^k$ define $W(w') \subset \mathbf{U}^{n-k}$ to be the set of continuations of w' to elements in W, i.e.,

$$W(w') = \{w'' \in \mathbf{U}^{n-k} : w'w'' \in W\}.$$

The following is an immediate corollary of Claim 3.10.

COROLLARY 3.12. $\mathbb{E}_{w' \in \mathbf{U}^k}[|W(w')|] \geq \delta|\mathbf{U}|^{n-k}$.

For $w' \in \mathbf{U}^k$ define $N(w')$ to be the set of distinct sets $L(w'w'')$,

$$N(w') = \{L(w'w'') : w'' \in W(w')\}.$$

Note that $|X| = \sum_{w' \in \mathbf{U}^k} |N(w')|$. We now prove lower bounds for the size of $N(w')$. These will then be used to prove lower bounds on $|X|$.

LEMMA 3.13. *Fix $w' \in \mathbf{U}^k$, and assume that $W(w') = \delta'|\mathbf{U}|^{n-k}$. Then*

$$|N(w')| \geq \delta' \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\alpha}\right)}.$$

Proof. Denote $|L(w')| = \beta'|\mathbf{U}|$, where $\beta' \geq \beta$ by condition (ii). Let $\widetilde{L} \in N(w')$ be some set. By condition (i) we know that $|\widetilde{L}| \leq \alpha|\mathbf{U}|$. Observe that if $L(w'w'') = \widetilde{L}$ for $w'' = x_{k+1}, \ldots, x_n \in W(w')$, then we must have $x_{k+1}, \ldots, x_n \in \widetilde{L}$. Moreover, by condition (iii) we must have that the number of elements of w'' that intersect $L(w')$ must be $\approx \beta'(n-k)$. Let m denote a possible number of elements of w'' that occur in $L(w')$. The number of sequences $w'' \in \mathbf{U}^{n-k}$ that contain exactly m elements in $L(w')$ and $n-k-m$ elements in $\widetilde{L} \setminus L(w')$ is given by

$$\binom{n-k}{m}|L(w')|^m(|\widetilde{L}| - |L(w')|)^{n-k-m} \leq \binom{n-k}{m}(\beta')^m(\alpha-\beta')^{n-k-m}|\mathbf{U}|^{n-k}.$$

Thus, the total number of $w'' \in W(w')$ for which $L(w'w'') = \widetilde{L}$ is bounded by

$$\begin{aligned}(5)\qquad &|\{w'' \in W(w') : L(w'w'') = \widetilde{L}\}| \\ &\leq \sum_{(\beta'-\gamma)(n-k) \leq m \leq (\beta'+\gamma)(n-k)} \binom{n-k}{m}(\beta')^m(\alpha-\beta')^{n-k-m}|\mathbf{U}|^{n-k}.\end{aligned}$$

On the other hand, we have that

$$\begin{aligned}|W(w')| = \delta'|\mathbf{U}|^{n-k} &= \delta' \sum_{m=0}^{n-k} \binom{n-k}{m}(\beta')^m(1-\beta')^{n-k-m}|\mathbf{U}|^{n-k} \\ (6)\qquad &\geq \delta' \sum_{m=(\beta'-\gamma)(n-k)}^{(\beta'+\gamma)(n-k)} \binom{n-k}{m}(\beta')^m(1-\beta')^{n-k-m}|\mathbf{U}|^{n-k}.\end{aligned}$$

Thus, the number of distinct sets $\widetilde{L} \in N(w')$ can be lower bounded by

$$\begin{aligned}|N(w')| &\geq \frac{|W(w')|}{\max_{\widetilde{L} \in N(w')} |\{w'' \in N(W') : L(w'w'') = \widetilde{L}\}|} \\ &\geq \delta' \frac{\sum_{m=(\beta'-\gamma)(n-k)}^{(\beta'+\gamma)(n-k)} \binom{n-k}{m}(\beta')^m(1-\beta')^{n-k-m}}{\sum_{m=(\beta'-\gamma)(n-k)}^{(\beta'+\gamma)(n-k)} \binom{n-k}{m}(\beta')^m(\alpha-\beta')^{n-k-m}}.\end{aligned}$$

As always, for any numbers $a_1, \ldots, a_t, b_1, \ldots, b_t > 0$ we have the bound

$$\frac{a_1 + \cdots + a_t}{b_1 + \cdots + b_t} \geq \min_i \frac{a_i}{b_i},$$

and we get the bound

$$\begin{aligned}(7)\qquad |N(w')| &\geq \delta' \min_{(\beta'-\gamma)(n-k) \leq m \leq (\beta'+\gamma)(n-k)} \left(\frac{1-\beta'}{\alpha-\beta'}\right)^{n-k-m} \\ &= \delta' \left(\frac{1-\beta'}{\alpha-\beta'}\right)^{(1-\beta'-\gamma)(n-k)}.\end{aligned}$$

Define a function $f : [0, \alpha) \to \mathbb{R}$ by $f(x) = (\frac{1-x}{\alpha-x})^{1-x}$. We show in Claim A.1 that f is monotone increasing. As $\alpha > \beta' \geq \beta$ we get that

$$\left(\frac{1-\beta'}{\alpha-\beta'}\right)^{1-\beta'} = f(\beta') \geq f(\beta) = \left(\frac{1-\beta}{\alpha-\beta}\right)^{1-\beta},$$

and hence

$$|N(w')| \geq \delta' \left(\frac{1-\beta'}{\alpha-\beta'}\right)^{(1-\beta')\left(\frac{1-\beta'-\gamma}{1-\beta'}\right)(n-k)} \tag{8}$$

$$\geq \delta' \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\beta'}\right)} \tag{9}$$

$$\geq \delta' \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\alpha}\right)}, \tag{10}$$

where in the last inequality we used that fact that $\beta' < \alpha$ since $L(w') \subset L(w'w'')$. ∎

We obtain as a corollary a lower bound on $|X|$.

Claim 3.14. $|X| \geq \delta |\mathbf{U}|^k (\frac{1-\beta}{\alpha-\beta})^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\alpha}\right)}$.

Proof. By Corollary 3.12 and Lemma 3.13 we have

$$\begin{aligned}
|X| &= \sum_{w' \in \mathbf{U}^k} |N(w')| \\
&\geq \sum_{w' \in \mathbf{U}^k} \frac{|W(w')|}{|\mathbf{U}|^{n-k}} \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\alpha}\right)} \\
&\geq \delta |\mathbf{U}|^{n-k} \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\alpha}\right)}.
\end{aligned}$$

∎

Combining Claims 3.11 and 3.14 we deduce the inequality

$$2^M \alpha^k \geq \delta \left(\frac{1-\beta}{\alpha-\beta}\right)^{(1-\beta)(n-k)\left(1-\frac{\gamma}{1-\alpha}\right)}. \tag{11}$$

We now fix parameters. Let $k = cn$, where $0 < c < 1$ is a fixed parameter. Denote $M = M_D(n, \varepsilon) = C \cdot n \log_2(1/\varepsilon)$, where a priori we know that $1 - o(1) \leq C \leq \log_2(e) \approx 1.44$. We will prove a lower bound on C.

We think of $n \gg 1$, where the parameters ε, c, C are fixed, and take $\delta = 1/n$. This gives the following quantities for α, β, γ:

$$\begin{aligned}
\alpha &= \varepsilon \left(1 + \frac{n}{|\mathbf{U}|}\right)(1 + 6\delta) = (1 + o(1))\varepsilon, \\
\beta &= \delta^{1/k} 2^{-M/k} = (1 + o(1))\varepsilon^{C/c}, \\
\gamma &= \sqrt{3 \ln(2/\delta)/(n-k)} = o(1).
\end{aligned}$$

Substituting the parameters into inequality (11), and taking $n \to \infty$, gives the following simplified form:

$$\left(\frac{1}{\varepsilon}\right)^C \varepsilon^c \geq \left(\frac{1-\varepsilon^{C/c}}{\varepsilon-\varepsilon^{C/c}}\right)^{(1-\varepsilon^{C/c})(1-c)}. \tag{12}$$

Note that for any given fixed value of ε, C, inequality (12) should hold for *any* value of $0 < c < 1$. Thus we are now left with the following analytical problem: For a given value of ε, what is the minimal value of C such that inequality (12) holds for all $0 < c < 1$?

3.4. Obtaining the lower bound from inequality (12). We start by noting that (12) is monotone in C; that is, if it holds for some C, it holds for all $C' > C$. This can be verified since the left-hand side is increasing with C while the right-hand side is decreasing (since the function $f(x) = (\frac{1-x}{\varepsilon-x})^{1-x}$ is monotone increasing in the range $[0, \varepsilon)$, as we show in Claim A.1). We thus define

$$C(\varepsilon) = \min\{C : \text{inequality (12) holds for } \varepsilon, C \text{ for all } 0 < c < 1\}.$$

We have the bound $M_D(n, \varepsilon) \geq C(\varepsilon) \cdot n \log_2(1/\varepsilon)$. It is easy to verify that taking limits $c \to 0$ or $c \to 1$ gives the bound $C(\varepsilon) \geq 1$, which we already knew from the information-theoretic lower bound. Thus, in order to get nontrivial lower bounds, we need to consider intermediate values of c.

For specific values of ε one can optimize over the value of C using a computer program. For example, see the following claim.

CLAIM 3.15. $C(1/2) > 1.1$.

Proof. One can verify by a direct calculation that inequality (12) is not satisfied for $\varepsilon = 1/2$, $C = 1.1$, and $c = 0.7$. Thus $C(1/2) > 1.1$. The best lower bound found empirically is $C(1/2) > 1.10213$. ∎

CLAIM 3.16. *For any $0 < \varepsilon < 1$ we have $C(\varepsilon) > 1$. Moreover, for small enough $\varepsilon > 0$ we have*

$$C(\varepsilon) \geq 1 + \Omega\left(\frac{1}{\log^2(1/\varepsilon)}\right).$$

Proof. We start by showing that $C(\varepsilon) > 1$ for any $0 < \varepsilon < 1$. Let $0 < c < 1$, to be fixed later, and define the function $f : [0, \varepsilon) \to \mathbb{R}$ by $f(x) = (\frac{1-x}{\varepsilon-x})^{1-x}$. Inequality (12) is equivalent to

$$(1/\varepsilon)^{\frac{C-c}{1-c}} \geq f(\varepsilon^{C/c}). \tag{13}$$

Note that $f(0) = 1/\varepsilon$ and that since $\varepsilon^{C/c} > 0$ we must have by Claim A.1 that $f(\varepsilon^{C/c}) > f(0)$. Thus we must have $C > 1$ in order to satisfy (13).

We now derive explicit bounds for small $\varepsilon > 0$. Set $c = (1 + \frac{a}{\log(1/\varepsilon)})^{-1}$, where $a > 0$ is a constant to be specified later. Note that this satisfies $\varepsilon^{1/c} = e^{-a} \cdot \varepsilon$. We will assume $\varepsilon > 0$ is small enough so that $c > 1/2$. This implies that $1 - c \geq \frac{a}{2\log(1/\varepsilon)}$.

We first apply Claim A.2. By the claim, there are absolute constants $c_1, c_2 > 0$ such that if we set a large enough so that $e^{-a} \leq c_1$, we get that $\varepsilon^{C/c} \leq \varepsilon^{1/c} \leq c_1\varepsilon$ and hence $f(\varepsilon^{C/c}) \geq 1/\varepsilon + c_2\varepsilon^{C/c}/\varepsilon^2$. Combining this with (13) we get that

$$(1/\varepsilon)^{\frac{C-1}{1-c}} \geq 1 + c_2\varepsilon^{C/c-1}.$$

We now use the fact that $\varepsilon^{1/c} = e^{-a}\varepsilon$ and that $C \leq 2$ (say) to infer there is an absolute constant $c_3 > 0$ such that

$$(1/\varepsilon)^{\frac{C-1}{1-c}} \geq 1 + c_3\varepsilon^{C-1}.$$

We may assume $c_3 \leq 1$. Taking logarithms from both sides, recalling that $\log(1+x) \geq x/2$ for $0 \leq x \leq 1$, and using the assumption that $1 - c \geq \Omega(1/\log(1/\varepsilon))$ we get that there is an absolute constant $c_4 > 0$ such that

$$C - 1 \geq \frac{c_4}{\log^2(1/\varepsilon)} \cdot \varepsilon^{C-1}.$$

Hence we conclude that there exists an absolute constant $c_5 > 0$ such that for small enough $\varepsilon > 0$,

$$C - 1 \geq \frac{c_5}{\log^2(1/\varepsilon)}. \qquad \square$$

4. Summary and open problems. In this work we gave the first nontrivial lower bound for the space requirements of any dynamic data structure for the approximate membership problem. The lower bound is $M_D(n,\varepsilon) \geq C(\varepsilon) \cdot n \log_2(1/\varepsilon)$ for $C(\varepsilon) = 1 + \Omega(1/\log^2(1/\varepsilon))$. We contrast this with the current best upper bound, $M_D(n,\varepsilon) \leq n \log_2(1/\varepsilon) + O(n)$. We leave as an intriguing open problem determining the true value of $M_D(n,\varepsilon)$. Another problem is to derive better algorithms (and lower bounds) in the case where both positive and negative errors are allowed; see e.g., [PR01].

4.1. Improved bounds via recursion. We note that one may use recursion of the argument we presented so far in order to derive an improved bound on $M_D(n,\varepsilon)$. The main claim which can be improved is Claim 3.6, which gives a bound on β in terms of a covering argument on the first k layers of the graph. We could use instead a recursive argument: first, derive a lower bound on $M_D(k,\varepsilon)$, and then use it to define β appropriately, i.e.,

$$\beta = \delta^{1/k} 2^{-M_D(k,\varepsilon)/k}.$$

This is a two-step recursive argument. A general r-step recursive argument entails choosing constants $0 < c_r < \cdots < c_1 < 1$ and performing the analysis for $\{k_i = c_i n\}$. It turns out that using a recursive argument improves the bounds we get using the nonrecursive approach, but only slightly. We performed a computer search for $\varepsilon = 1/2$ for a recursive sequence $c_1 > \cdots > c_r$ that will give the best result. We obtained the bound $C(1/2) \geq 1.13$, compared with $C(1/2) \geq 1.102$, which can be obtained by a nonrecursive argument.

Appendix A. Proof of technical claims. Let $0 < \alpha < 1$ and define $f : [0,\alpha) \to \mathbb{R}$ by $f(x) = (\frac{1-x}{\alpha-x})^{1-x}$.

CLAIM A.1. *f is monotone increasing.*

Proof. Let $g(x) = \ln(f(x)) = (1-x)(\ln(1-x) - \ln(\alpha - x))$. It is sufficient to prove that g is monotone increasing. We have

$$\begin{aligned} g'(x) &= \ln(\alpha - x) - \ln(1-x) - 1 + \frac{1-x}{\alpha - x} \\ &= -\ln\left(\frac{1-x}{\alpha-x}\right) - 1 + \frac{1-x}{\alpha-x}. \end{aligned}$$

For any $z > 0$ we have $e^z > 1 + z$. Thus for any $y > 1$ we have $\ln(y) < y - 1$. Set $y = \frac{1-x}{\alpha-x} > 1$. We have

$$g'(x) = -\ln(y) - 1 + y > 0.$$

Hence g is monotone increasing, and so is f. $\square$

We next lower bound the value of f near zero.

CLAIM A.2. *There exist absolute constants $c_1, c_2 > 0$ such that the following holds. For any $\alpha \le 1/2$ and any $0 \le x \le c_1\alpha$ we have*

$$f(x) \ge 1/\alpha + c_2 x/\alpha^2.$$

Proof. We have $f(0) = 1/\alpha$. As $f(x) = \exp(g(x))$ we have $f'(x) = f(x)g'(x)$ and $f''(x) = f(x)(g'(x)^2 + g''(x))$. We have

$$g'(x) = -\ln\left(\frac{1-x}{\alpha - x}\right) - 1 + \frac{1-x}{\alpha - x},$$
$$g''(x) = \frac{(1-\alpha)^2}{(1-x)(\alpha-x)^2}.$$

It is straightforward to verify that for $\alpha \le 0.1$ we have $f'(0) = \Omega(1/\alpha^2)$ and that for any $0 \le x \le \alpha/2$ we have $f''(x) \le O(1/\alpha^3)$. Hence by Taylor's theorem we have for all $0 \le x \le \alpha/2$ that

$$f(x) \ge 1/\alpha + \Omega(x/\alpha^2) - O(x^2/\alpha^3).$$

Thus there exist constants $c_1, c_2 > 0$ such that as long as $x \le c_1\alpha$ the second term dominates the third term; that is,

$$f(x) \ge 1/\alpha + c_2 x/\alpha^2. \qquad \square$$

REFERENCES

[ANS10] Y. ARBITMAN, M. NAOR, AND G. SEGEV, *Backyard cuckoo hashing: Constant worst-case operations with a succinct representation*, in Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS '10), IEEE Computer Society, Washington, DC, 2010, pp. 787–796.

[Blo70] B. H. BLOOM, *Space/time trade-offs in hash coding with allowable errors*, Commun. ACM, 13 (1970), pp. 422–426.

[BM03] A. BRODER AND M. MITZENMACHER, *Network applications of bloom filters: A survey*, Internet Math., 1 (2003), pp. 485–509.

[CFG+78] L. CARTER, R. FLOYD, J. GILL, G. MARKOWSKY, AND M. WEGMAN, *Exact and approximate membership testers*, in Proceedings of the Tenth Annual ACM Symposium on Theory of Computing (STOC '78), ACM, New York, 1978, pp. 59–65.

[DP08] M. DIETZFELBINGER AND R. PAGH, *Succinct data structures for retrieval and approximate membership*, in Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP '08), Part I, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 385–396.

[MP07] Y. MATIAS AND E. PORAT, *Efficient pebbling for list traversal synopses with application to program rollback*, Theoret. Comput. Sci., 379 (2007), pp. 418–436.

[MV08] M. MITZENMACHER AND S. P. VADHAN, *Why simple hash functions work: Exploiting the entropy in a data stream*, in Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '08), SIAM, Philadelphia, 2008, pp. 746–755.

[Por09] E. PORAT, *An optimal bloom filter replacement based on matrix solving*, in Proceedings of the Fourth International Computer Science Symposium in Russia on Computer Science—Theory and Applications (CSR '09), Springer-Verlag, Berlin, Heidelberg, 2009, pp. 263–273.

[PPR05] A. PAGH, R. PAGH, AND S. S. RAO, *An optimal bloom filter replacement*, in Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '05), SIAM, Philadelphia, ACM, New York, 2005, pp. 823–829.

[PR01] R. Pagh and F. F. Rodler, *Lossy dictionaries*, in Proceedings of the 9th Annual European Symposium on Algorithms (ESA '01), Springer-Verlag, New York, 2001, pp. 300–311.

[RR03] R. Raman and S. S. Rao, *Succinct dynamic dictionaries and trees*, in Proceedings of the 30th International Conference on Automata, Languages, and Programming (ICALP '03), Springer-Verlag, Berlin, Heidelberg, 2003, pp. 357–368.