Certificates in Data Structures

Yaoyu Wang *† Yitong Yin*‡

Abstract

We study certificates in static data structures. In the cell-probe model, certificates are the cell probes which can uniquely identify the answer to the query. As a natural notion of nondeterministic cell probes, lower bounds for certificates in data structures immediately imply deterministic cell-probe lower bounds. In spite of this extra power brought by nondeterminism, we prove that two widely used tools for cell-probe lower bounds: richness lemma of Miltersen *et al.* [10] and direct-sum richness lemma of Pătraşcu and Thorup [17], both hold for certificates in data structures with even better parameters. Applying these lemmas and adopting existing reductions, we obtain certificate lower bounds for a variety of static data structure problems. These certificate lower bounds are at least as good as the highest known cell-probe lower bounds for the respective problems. In particular, for approximate near neighbor (ANN) problem in Hamming distance, our lower bound improves the state of the art. When the space is strictly linear, our lower bound for ANN in d-dimensional Hamming space becomes $t = \Omega(d)$, which along with the recent breakthrough for polynomial evaluation of Larsen [8], are the only two $t = \Omega(d)$ lower bounds ever proved for any problems in the cell-probe model.

1 Introduction

In static data structure problems, a database is preprocessed to form a table according to certain encoding scheme, and upon each query to the database, an algorithm (decision tree) answers the query by adaptively probing the table cells. The complexity of this process is captured by the cell-probe model for static data structures. Solutions in this model are called cell-probing schemes.

The cell-probe model plays a central role in studying data structure lower bounds. The existing cell-probe lower bounds for static data structure problems can be classified into the following three categories according to the techniques they use and the highest possible lower bounds supported by these techniques:

- Lower bounds implied by asymmetric communication complexity: Classic techniques introduced in the seminal work of Miltersen *et al.* [10] see a cell-probing scheme as a communication protocol between the query algorithm and the table, and the cell-probe lower bounds are implied by the asymmetric communication complexity lower bounds which are proved by the *richness lemma* or round eliminations. In the usual setting that both query and data items are points from a d-dimensional space, the highest time lower bound that can be proved in this way is $t = \Omega\left(\frac{d}{\log s}\right)$ with a table of s cells. This bound is a barrier for the technique, because a matching upper bound can always be achieved by communication protocols.

*State Key Laboratory for Novel Software Technology, Nanjing University, China.
†Email: yaoyu.wang.nju@gmail.com.
‡Supported by NSFC grants 61272081, 61003023 and 61321491. Email: yinyt@nju.edu.cn.

- Lower bounds proved by self-reduction using direct-sum properties: The seminal works of Pătraşcu and Thorup [16, 17] introduce a very smart idea of many-to-one self-reductions, using which and by exploiting the direct-sum nature of problems, higher lower bounds can be proved for a near-linear space. The highest lower bounds that can be proved in this way is $t = \Omega\left(d/\log \frac{sw}{n}\right)$ with a table of s cells each containing w bits. Such lower bounds grow differently with near-linear space and polynomial space, which is indistinguishable in the communication model.

- Higher lower bounds for linear space: A recent breakthrough of Larsen [8] uses a technique refined from the cell sampling technique of Panigrahy *et al.* [11, 12] to prove an even higher lower bound for the polynomial evaluation problem. This lower bound behaves as $t = \Omega(d)$ when the space is strictly linear. This separates for the first time between the cell-probe complexity with linear and near-linear spaces, and also achieves the highest cell-probe lower bound ever known for any data structure problems.

In this paper, we consider an even stronger model: *certificates* in static data structures. A query to a database is said to have certificate of size t if the answer to the query can be uniquely identified by the contents of t cells in the table. This very natural notion represents the nondeterministic computation in cell-probe model and is certainly a lower bound to the complexity of deterministic cell-probing schemes. This nondeterministic model has been explicitly considered before in a previous work [19] of one of the authors of the current paper.

Surprisingly, in spite of the seemingly extra power brought by the nondeterminism, the highest cell-probe lower bound to date is in fact a certificate lower bound [8]. Indeed, we conjecture that for typical data structure problems, especially those hard problems, *the complexity of certifying the answer should dominate that of computing the answer.*[1] This belief has been partially justified in [19] by showing that a random static data structure problem is hard nondeterministically. In this paper, we further support this conjecture by showing that several mainstream techniques for cell-probe lower bounds in fact can imply as good or even higher certificate lower bounds.

1.1 Our contributions

We make the following contributions:

1. We prove a richness lemma for certificates in data structures, which improves the classic richness lemma for asymmetric communication complexity of Miltersen *et al.* [10] in two ways: (1) when applied to prove data structure lower bounds, our richness lemma implies lower bounds for a stronger *nondeterministic* model; and (2) our richness lemma achieves better parameters than the classic richness lemma and may imply higher lower bounds.

2. We give a scheme for proving certificate lower bounds using a similar direct-sum based self-reduction of Pătraşcu and Thorup [17]. The certificate lower bounds obtained from our scheme is at least as good as before when the space is near-linear or polynomial. And for strictly linear space, our technique may support superior lower bounds, which was impossible for the direct-sum based techniques before.

[1] Interestingly, the only known exception to this conjecture is the predecessor search problem whose cell-probe complexity is a mild super-constant while the queries can be easily certified with constant cells in a sorted table.

problem	certificate lower bound proved here	highest known cell-probe lower bound
bit-vector retrieval	$t = \Omega\left(\frac{m \log n}{\log s}\right)$	not known
lopsided set disjointness (LSD)	$t = \Omega\left(\frac{m \log n}{\log s}\right)$	$t = \Omega\left(\frac{m \log n}{\log s}\right)$ [1, 10, 15]
approximate near neighbor (ANN) in Hamming space	$t = \Omega\left(d/\log \frac{sw}{nd}\right)^{\diamond}$	$t = \Omega\left(d/\log \frac{sw}{n}\right)^{\star}$ [11, 17]
partial match (PM)	$t = \Omega\left(d/\log \frac{sw}{n}\right)^{\star}$	$t = \Omega\left(d/\log \frac{sw}{n}\right)^{\star}$ [11, 17]
3-ANN in ℓ_∞	$t = \Omega\left(d/\log \frac{sw}{n}\right)^{\star}$	$t = \Omega\left(d/\log \frac{sw}{n}\right)^{\star}$ [17]
reachability oracle 2D stabbing 4D range reporting	$t = \Omega\left(\log n/\log \frac{sw}{n}\right)^{\star}$	$t = \Omega\left(\log n/\log \frac{sw}{n}\right)^{\star}$ [15]
2D range counting	$t = \Omega\left(\log n/\log \frac{sw}{n}\right)^{\star}$	$t = \Omega\left(\log n/\log \frac{sw}{n}\right)^{\star}$ [13, 15]
approximate distance oracle	$t = \Omega\left(\frac{\log n}{\alpha \log(s \log n/n)}\right)^{\star}$	$t = \Omega\left(\frac{\log n}{\alpha \log(s \log n/n)}\right)^{\star}$ [18]

$\star$: lower bound which grows differently with near-linear and polynomial space;
$\diamond$: lower bound which grows differently with linear, near-linear, and polynomial space.

Table 1: Certificate lower bounds proved in this paper.

3. By applying these techniques, adopting the existing reductions, and modifying the reductions in the communication model to be model-independent, we prove certificate lower bounds for a variety of static data structure problems, listed in Table 1. All these certificate lower bounds are at least as good as the highest known cell-probe lower bounds for the respective problems. And for approximate near neighbor (ANN), our $t = \Omega\left(d/\log \frac{sw}{nd}\right)$ lower bound improves the state of the art. When the space $sw = O(nd)$ is strictly linear, our lower bound for ANN becomes $t = \Omega(d)$, which along with the recent breakthrough for polynomial evaluation [8], are the only two $t = \Omega(d)$ lower bounds ever proved for any problems in the cell-probe model.

1.2 Related work

The richness lemma, along with the round elimination lemma, for asymmetric communication complexity was introduced in [10]. The richness lemma was later widely used, for example in [2, 3, 7, 9], to prove lower bounds for high dimensional geometric problems, e.g. nearest neighbor search. In [1, 15], a generalized version of richness lemma was proved to imply lower bounds for (Monte Carlo) randomized data structures. A direct-sum richness theorem was first proved in the conference version of [17]. Similar but less involved many-to-one reductions were used in [15] and [18] for proving lower bounds for certain graph oracles.

The idea of cell sampling was implicitly used in [13] and independently in [11]. This novel technique was later fully developed in [12] for high dimensional geometric problems and in [8] for polynomial evaluation. The lower bound in [8] actually holds for nondeterministic cell probes, i.e. certificates. The nondeterministic cell-probe complexity was studied for dynamic data structure problems in [5] and for static data structure problems in [19].

2 Certificates in data structures

A data structure problem is a function $f : X \times Y \to Z$ with two domains X and Y. We call each $x \in X$ a **query** and each $y \in Y$ a **database**, and $f(x, y) \in Z$ specifies the result of query x on database y. A code $T : Y \to \Sigma^s$ with an alphabet $\Sigma = \{0, 1\}^w$ transforms each database $y \in Y$ to a **table** $T_y = T(y)$ of s **cells** each containing w bits. We use $[s] = \{1, 2, \ldots, s\}$ to denote the set of indices of cells, and for each $i \in [s]$, we use $T_y(i)$ to denote the content of the i-th cell of table T_y.

A data structure problem is said to have (s, w, t)**-certificates**, if any database can be stored in a table of s cells each containing w bits, so that the result of each query can be uniquely determined by contents of at most t cells. Formally, we have the following definition.

Definition 1 *A data structure problem $f : X \times Y \to Z$ is said to have (s, w, t)-**certificates**, if there exists a code $T : Y \to \Sigma^s$ with an alphabet $\Sigma = \{0, 1\}^w$, such that for any query $x \in X$ and any database $y \in Y$, there exists a subset $P \subseteq [s]$ of cells with $|P| = t$, such that for any database $y' \in Y$, we have $f(x, y') = f(x, y)$ if $T_{y'}(i) = T_y(i)$ for all $i \in P$.*

Because certificates represent nondeterministic computation in data structures, it is obvious that it has stronger computational power than cell-probing schemes.

Proposition 2 *For any data structure problem f, if there is a cell-probing scheme storing every database in s cells each containing w bits and answering every query within t cell-probes, then f has (s, w, t)-certificates.*

Data structure certificates can be equivalently formulated as proof systems as well as certificates in decision trees of partial functions.

As proof systems. In a previous work [19], an equivalent formulation of data structure certificates as proof systems is used. A data structure problem $f : X \times Y \to Z$ has (s, w, t)-certificates if and only if there exist a code $T : Y \to \Sigma^s$ with an alphabet $\Sigma = \{0, 1\}^w$ and a verifier $V : \{0, 1\}^* \to Z \cup \{\perp\}$ where $\perp$ is a special symbol not in Z indicating the failure of verification, so that for any query $x \in X$ and any database $y \in Y$, the followings are satisfied:

- Completeness: $\exists P \subseteq [s]$ with $|P| = t$ such that $V(x, \langle i, T_y(i)\rangle_{i \in P}) = f(x, y)$;
- Soundness: $\forall P' \subseteq [s]$ with $|P'| = t$, $V(x, \langle i, T_y(i)\rangle_{i \in P'}) \in \{f(x, y), \perp\}$;

where $\langle i, T_y(i)\rangle_{i \in P}$ denotes the sequence of pairs $\langle i, T_y(i)\rangle$ for all $i \in P$.

As certificates in decision trees. Certificate is a well-known notion is studies of decision tree complexity (see [4] for a survey). A certificate in a Boolean function $h : \{0, 1\}^n \to \{0, 1\}$ for an input $x \in \{0, 1\}^n$ is a subset $i_1, i_2, \ldots, i_t \in [n]$ of t bits in x such that for every $x' \in \{0, 1\}^n$ satisfying that $x'(i_j) = x(i_j)$ for all $1 \le j \le t$, it holds that $h(x) = h(x')$. And the certificate complexity of h, denoted by $C(h)$, is the minimum number of bits in a certificate in the worst-case of input x. The certificates and certificate complexity $C(h)$ can be naturally generalized to partial function $h : \Sigma^s \to Z$ with non-Boolean domain Σ and range Z.

Given a data structure problem $f : X \times Y \to Z$, and a code $T : Y \to \Sigma^s$ with an alphabet $\Sigma = \{0, 1\}^w$, for each query $x \in X$, the function f can be naturally transformed into a partial function $f^T_x : \Sigma^s \to Z$ so that $f^T_x(T_y) = f(x, y)$ for every database $y \in Y$ and f^T_x is not defined elsewhere.

It is easy to verify that a data structure problem $f : X \times Y \to Z$ has (s, w, t)-certificates if and only if there exists a code $T : Y \to \Sigma^s$ with an alphabet $\Sigma = \{0,1\}^w$ such that $\max_{x \in X} C(f_x^T) \leq t$, where $C(f_x^T)$ is the certificate complexity of the partial function $f_x^T : \Sigma^s \to Z$.

3 The richness lemma

From now on, we focus on the decision problems where the output is either 0 or 1. A data structure problem $f : X \times Y \to \{0,1\}$ can be naturally treated as an $|X| \times |Y|$ matrix whose rows are indexed by queries $x \in X$ and columns are indexed by data $y \in Y$. The entry at the x-th row and y-th column is $f(x, y)$. For $\xi \in \{0,1\}$, we say f has a **monochromatic ξ-rectangle** of size $k \times \ell$ if there is a combinatorial rectangle $A \times B$ with $A \subseteq X, B \subseteq Y, |A| = k$ and $|B| = \ell$, such that $f(x, y) = \xi$ for all $(x, y) \in A \times B$. A matrix f is said to be (u, v)**-rich** if at least v columns contain at least u 1-entries. The following richness lemma for cell-probing schemes is introduced in [10].

Lemma 3 (Richness Lemma [10]) *Let f be a (u, v)-rich problem. If f has an (s, w, t)-cell-probing scheme, then f contains a monochromatic 1-rectangle of size $\frac{u}{2^{t \log s}} \times \frac{v}{2^{wt + t \log s}}$.*

In [10], the richness lemma is proved for asymmetric communication protocols. A communication protocol between two parties Alice and Bob is called an $[A, B]$-protocol if Alice sends Bob at most A bits and Bob sends Alice at most B bits in total in the worst-case. The richness lemma states that existence of $[A, B]$-protocol for a (u, v)-rich problem f implies a submatrix of dimension $\frac{u}{2^A} \times \frac{v}{2^{A+B}}$ containing only 1-entries. An (s, w, t)-cell-probing scheme can imply an $[A, B]$-protocol with $A = t \log s$ and $B = wt$, so the above richness lemma for the cell-probing schemes follows.

3.1 Richness lemma for certificates

We prove a richness result for data structure certificates, with even a better reliance on parameters.

Lemma 4 (Richness Lemma for data structure certificates) *Let f be a (u, v)-rich problem. If f has (s, w, t)-certificates, then f contains a monochromatic 1-rectangle of size $\frac{u}{\binom{s}{t}} \times \frac{v}{\binom{s}{t} 2^{wt}}$.*

Remark. Note that we always have $\log \binom{s}{t} = t \log \frac{s}{t} + O(t) \leq t \log s$. The bound in Lemma 4 is at least as good as the bound in classic richness lemma, even though now it is proved for nondeterministic computation. When s and t are close to each other, the bound in Lemma 4 is substantially better than that of classic richness lemma. Later in Section 4, this extra gain is used in direct-sum reductions introduced in [17] to achieve better time lower bounds for linear or near-linear space which match or improve state of the art. It is quite shocking to see all these achieved through a very basic reduction to the 1-probe case to be introduced later.

The classic richness lemma for asymmetric communication protocol is proved by a halving argument. Due to determinism of communication protocols (and cell-probing schemes), the combinatorial rectangle obtained from halving the universe are *disjoint*. This disjointness no longer holds for the rectangles obtained from certificates because of nondeterminism. We resolve this issue by exploiting combinatorial structures of rectangles obtained from data structure certificates.

The following preparation lemma is a generalization of the averaging principle.

Lemma 5 *Let $\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_r \subset 2^V$ be partitions of V satisfying $|\mathcal{P}_i| \le k$ for every $1 \le i \le k$. There must exist a $y \in V$ such that $|\mathcal{P}_i(y)| \ge \frac{|V|}{rk}$ for all $1 \le i \le r$, where $\mathcal{P}_i(y)$ denotes the partition block $B \in \mathcal{P}_i$ containing y.*

Proof: The lemma is proved by the probabilistic method. Let y be uniformly chosen from V. Fix an arbitrary order of partition blocks for each partition $\mathcal{P}_i$. Let w_{ij} be the cardinality of the j-th block in $\mathcal{P}_i$. Obviously the probability of $\mathcal{P}_i(y)$ being the j-th block in $\mathcal{P}_i$ is $\frac{w_{ij}}{|V|}$. By union bound, the probability that $|\mathcal{P}_i(y)| < w$ is bounded by $\sum_{j:w_{ij}<w} \frac{w_{ij}}{|V|} < |\{j : w_{ij} < w\}| \frac{w}{|V|}$. Since $|\mathcal{P}_i| \le k$, for every i there are at most k many such j satisfying that $w_{ij} < w$, thus $\Pr\left[|\mathcal{P}_i(y)| < \frac{|V|}{rk}\right] < k \cdot \frac{|V|/rk}{|V|} = \frac{1}{r}$. Applying union bound again for all $\mathcal{P}_i$, we have $\Pr\left[\exists 1 \le i \le r, |\mathcal{P}_i(y)| < \frac{|V|}{rk}\right] < 1$, which means there exists a $y \in V$ such that $|\mathcal{P}_i(y)| \ge \frac{|V|}{rk}$ for all $1 \le i \le r$. ∎

We first prove the richness lemma for the 1-probe case.

Lemma 6 *Let f be a (u, v)-rich problem. If f has $(s, w, 1)$-certificates, then f contains a monochromatic 1-rectangle of size $\frac{u}{s} \times \frac{v}{s \cdot 2^w}$.*

Proof: Let $T : Y \to \Sigma^s$ where $\Sigma = \{0,1\}^w$ be the code in the $(s, w, 1)$-certificates for f. Let $V \subseteq Y$ denote the set of v columns of f that each contains at least u 1-entries. For each cell $1 \le i \le s$, an equivalence relation $\sim_i$ on databases in V can be naturally defined as follows: for any $y, y' \in V$, $y \sim_i y'$ if $T_y(i) = T_{y'}(i)$, that is, if databases y and y' look same in the i-th cell. Let $\mathcal{P}_i$ denote the partition induced by the equivalence relation $\sim_i$. Each partition $\mathcal{P}_i$ classifies the databases in V according to the content of the i-th cell. Obviously $|\mathcal{P}_i| \le 2^w$, because the content of a cell can have at most $|\Sigma| = 2^w$ possibilities, and we also have $\mathcal{P}_i(y) = \{y' \in V \mid T_{y'}(i) = T_y(i)\}$ being the set of databases indistinguishable from y by looking at the i-th cell, where $\mathcal{P}_i(y)$ denotes the partition block $B \in \mathcal{P}_i$ containing y. By Lemma 5, there always exists a bad database $y \in V$ such that $|\mathcal{P}_i(y)| \ge \frac{|V|}{s \cdot 2^w} = \frac{v}{s \cdot 2^w}$ for all $1 \le i \le s$.

For each database $y \in V$, let $X_1(y) = \{x \in X \mid f(x, y) = 1\}$ denote the set of positive queries on database y, and for a subset $A \subseteq V$ of databases, let $X_1(A) = \bigcap_{y \in A} X_1(y)$ denote the set of queries which are positive on all databases in A. Note that $X_1(y)$ and $X_1(A)$ are the respective 1-preimages of Boolean functions $f(\cdot, y)$ and $\bigwedge_{y \in A} f(\cdot, y)$. By definition, it is easy to see that $X_1(A) \times A$ is a monochromatic 1-rectangle for any $A \subseteq V$.

Claim: For any $y \in V$, it holds that $X_1(y) = \bigcup_{1 \le i \le s} X_1(\mathcal{P}_i(y))$.

It is easy to see the direction $\bigcup_{1 \le i \le s} X_1(\mathcal{P}_i(y)) \subseteq X_1(y)$ holds because $X_1(A) \subseteq X_1(y)$ for any A containing y and clearly $y \in \mathcal{P}_i(y)$. So we only need to prove the other direction. Since f has $(s, w, 1)$-certificates, for any positive query x on database y (i.e. any $x \in X_1(y)$), there is a cell i such that all databases y' indistinguishable from y by looking at the i-th cell (i.e. all $y' \in \mathcal{P}_i(y)$) answer the query x positively (i.e. $f(x, y') = f(x, y) = 1$), which gives $x \in X_1(\mathcal{P}_i(y))$ by definition of $X_1(A)$. This proves the direction $X_1(y) \subseteq \bigcup_{i \in [s]} X_1(\mathcal{P}_i(y))$.

Consider the bad database $y \in V$ satisfying $|\mathcal{P}_i(y)| \ge \frac{|V|}{s \cdot 2^w} = \frac{v}{s \cdot 2^w}$ for all $1 \le i \le s$. Due to the above claim, we have

$$u \le |X_1(y)| = \left| \bigcup_{1 \le i \le s} X_1(\mathcal{P}_i(y)) \right| \le \sum_{1 \le i \le s} |X_1(\mathcal{P}_i(y))| .$$

By averaging principle, there exists a cell i such that $|X_1(\mathcal{P}_i(y))| \geq \frac{u}{s}$. This gives us a monochromatic 1-rectangle $X_1(\mathcal{P}_i(y)) \times \mathcal{P}_i(y)$ of size at least $\frac{u}{s} \times \frac{v}{s \cdot 2^w}$. ∎

The richness lemma for general case can be derived from the 1-probe case by a one-line reduction.

Lemma 7 *If a data structure problem f has (s, w, t)-certificates, then f has $\left(\binom{s}{t}, w \cdot t, 1\right)$-certificates.*

Proof: Store every t-combination of cells with a new table of $\binom{s}{t}$ cells each of $w \cdot t$ bits. ∎

3.2 Applications

We apply our richness lemma to two fundamental data structure problems: the bit-vector retrieval problem, and the lopsided set disjointness (LSD) problem. We prove certificate lower bounds matching the cell-probing scheme upper bounds, which shows that for these fundamental data structure problems, answering queries is as hard as certifying them.

Bit-vector retrieval. We consider the following fundamental problem: a database y is a vector of n bits, a query x specifies m indices, and the answer to the query returns the contents of these queried bits in the bit vector y. Although is fundamental in database and information retrieval even judging by a glance, this problem has not been very well studied before (for a reason which we will see next). We call this problem the **bit-vector retrieval** problem. A naive solution is to explicitly store the bit-vector and access the queried bits directly, which gives an bit-probing scheme using n bits and answering each query with m bits. A natural and important question is: can we substantially reduce the time cost by using a more sophisticated data structure with a tolerable overhead on space usage and allowing probing cells instead of bits? We shall see this is impossible in any realistic setting by showing a certificate lower bound.

We study a decision version of the bit-vector retrieval problem, namely the **bit-vector testing** problem. Let $Y = \{0,1\}^n$ and $X = [n]^m \times \{0,1\}^m$. Each database $y \in Y$ is still an n-bit vector, and each query $x = (u, v) \in X$ consists of two parts: a tuple $u \in [n]^m$ of m positions and a prediction $v \in \{0,1\}^m$ of the contents of these positions. For $y \in \{0,1\}^n$ and $u \in [n]^m$, we use $y(u)$ to denote the m-tuple $(y(u_1), y(u_2), \ldots, y(u_m))$. The bit-vector testing problem $f : X \times Y \to \{0,1\}$ is then defined as that for any $x = (u, v) \in X$ and any $y \in Y$, $f(x, y)$ indicates whether $y(u) = v$.

Proposition 8 *The bit-vector testing problem f is $(n^m, 2^n)$-rich and every $M \times N$ monochromatic 1-rectangles in f must have $M \leq (n - \log N)^m$.*

Proof: We use the notation in the proof of Lemma 6: we use $X_1(y)$ to denote set of positive queries on database y and $X_1(A)$ to denote the set of queries positive on all databases in $A \subset Y$. Note that $X_1(y)$ contains all the rows at which column y has 1-entries. It holds that $|Y| = n^m$ and for every $y \in Y$, we have $|X_1(y)| = |\{(u, v) \in [n]^m \times \{0,1\}^m \mid y(u) = v\}| = n^m$, thus f is $(n^m, 2^n)$-rich.

For any set $A \subseteq Y$, observe that $|X_1(A)| = |\{u \in [n]^m \mid \forall y, y' \in A, y(u) = y'(u)\}|$, i.e. $|X_1(A)|$ is the number of such m-tuples of indices over which all bit-vectors in A are identical. Let S denote the largest $S \subseteq [n]$ such that for every $i \in S$, $y(i)$ is identical for all $y \in A$. It is easy to see that $|X_1(A)| = |S|^m$ and $|A| \leq 2^{n-|S|}$, therefore it holds that $|X_1(A)| \leq (n - \log |A|)^m$. Note that $X_1(A) \times A$ is precisely the maximal 1-rectangle with the set of columns A. Letting $N = |A|$, we prove that every $M \times N$ 1-rectangle must have $M \leq (n - \log N)^m$. ∎

Theorem 9 *If the bit-vector testing problem has (s, w, t)-certificates, then for any $0 < \delta < 1$, we have either $t \geq \frac{n^{1-\delta}}{w+\log s}$ or $t \geq \frac{\delta m \log n}{\log s}$.*

Proof: Due to Proposition 8, the problem is $(n^m, 2^n)$-rich, and hence by Lemma 4, if it has (s, w, t)-certificates, then it contains a 1-rectangle of size $\frac{n^m}{\binom{s}{t}} \times 2^{n-wt-t\log\binom{s}{t}}$. As $\binom{s}{t} \leq s^t$, so we have a 1-rectangle of size $\frac{n^m}{s^t} \times 2^{n-wt-t\log s}$, which by Proposition 8, requires that $\frac{n^m}{s^t} \leq (wt + t\log s)^m$. For any $0 < \delta < 1$, if $t < \frac{n^{1-\delta}}{w+\log s}$, then $t \geq \frac{\delta m \log n}{\log s}$. ∎

A standard setting for data structure is the lopsided case, where query is significantly shorter than database. For this case, the above theorem has the following corollary.

Corollary 10 *Assuming $m = n^{o(1)}$, if the bit-vector testing problem has (s, w, t)-certificates for $w \leq n^{1-\delta}$ where $\delta > 0$ is an arbitrary constant, then $t = \Omega\left(\frac{m \log n}{\log s}\right)$.*

With any polynomial space $s = n^{O(1)}$ and a wildly relaxed size of cell $n^{1-\delta}$, the above bound matches the naive solution of directly retrieving m bits, implying that the fundamental problem of retrieving part of a bit vector cannot be made any easier in a general setting, because queries are hard to certify.

Lopsided set disjointness. The set disjointness problem plays a central role in communication complexity and complexity of data structures. Assuming a data universe $[N]$, the input domains are $X = \binom{[N]}{m}$ and $Y = \binom{[N]}{n}$ where $m \leq n < \frac{N}{2}$. For each query set $x \in X$ and data set $y \in Y$, the set disjointness problem $f(x, y)$ returns a bit indicating the emptyness of $x \cap y$. The following proposition is implicit in [10].

Proposition 11 (Milersen *et al.* [10]) *The set disjointness problem f is $\left(\binom{N-n}{m}, \binom{N}{n}\right)$-rich, and for every $n \leq u \leq N$, any monochromatic 1-rectangle in f of size $M \times \binom{u}{n}$ must have $M \leq \binom{N-u}{m}$.*

Proof: We use the notation in the proof of Lemma 6: let $X_1(y)$ denote set of positive queries on database y and $X_1(A)$ denote the set of queries positive on all databases in $A \subset Y$. $X_1(y)$ contains all the rows at which column y has 1-entries. It holds that $|Y| = \binom{N}{n}$ and for every set $y \in Y$ with $|y| = n$, we have $|X_1(y)| = |\{x \mid x \subset [N], |x| = m, x \cap y = \emptyset\}| = \binom{N-n}{m}$, thus f is $\left(\binom{N-n}{m}, \binom{N}{n}\right)$-rich.

For any set $A \subseteq Y$ with $|A| = \binom{u}{n}$, let $y' = \bigcup_{y \in A} y$. We have $|y'| \geq u$. For $X_1(A) = \{x \mid \forall y \in Y, x \cap y = \emptyset\}$, we have $|X_1(A)| \leq \binom{N-|y'|}{m} \leq \binom{N-u}{m}$. Thus we get the conclusion. ∎

Applying the above proposition and Lemma 4, we have the following certificate lower bound.

Theorem 12 *If the set disjointness problem has (s, w, t)-certificates, then for any $0 < \delta < 1$, we have either $t \geq \frac{n^{1-\delta}}{w+\log s}$ or $t \geq \frac{\delta m(\log n - o(1))}{\log s}$.*

Proof: Due to Proposition 11, the problem is $\left(\binom{N-n}{m}, \binom{N}{n}\right)$-rich, and hence by Lemma 4, if it has (s, w, t)-certificates, then it contains a 1-rectangle of size $\frac{\binom{N-n}{m}}{\binom{s}{t}} \times \frac{\binom{N}{n}}{\binom{s}{t}2^{wt}}$. As $\binom{s}{t} \leq s^t$, so we have a 1-rectangle of size $\binom{N-n}{m}/2^{t\log s} \times \binom{N}{n}/2^{wt+t\log s}$. Let $a = t\log s$ and $b = wt$. Let M, u

denote the parameters in Proposition 11 respectively, so $M = \binom{N-n}{m}/2^a$ and $\binom{u}{n} = \binom{N}{n}/2^{a+b}$. Let $k = (N-n)/2^{a/m}$. Since $\binom{k}{m} \le \binom{N-n}{m}/2^a \le \binom{N-u}{m}$ by Proposition 11, we have $k \le N-u$, which leads to $u \le N-k$. Now we have $\binom{N}{n}/2^{a+b} = \binom{u}{n} \le \binom{N-k}{n}$ and therefore $2^{a+b} \ge \binom{N}{n}/\binom{N-k}{n} > (\frac{N}{N-k})^n > (1+k/N)^n = (1+(N-n)/2^{a/m}N)^n \ge (1+2^{-a/m-1})^n$. By taking logarithm, we have $a+b \ge n\log(1+2^{-a/m-1}) > n \cdot 2^{-a/m-1}$. If $a+b < n^{1-\delta}$ for any δ, then $2^{a/m+1} \ge n^\delta$, thus $a \ge \delta m(\log n - o(1))$. Replacing a, b with $t\log s, wt$ respectively, we get the conclusion. ∎

This certificate lower bound matches the well-known cell-probe lower bound for set-disjointness [10, 15]. The most interesting case of the problem is the lopsided case where $m = n^{o(1)}$. A calculation gives us the following corollary.

Corollary 13 *Assume $m = n^{o(1)}$ and $\alpha n \le N \le n^c$ for arbitrary constants $\alpha, c > 1$. If the set disjointness problem has (s, w, t)-certificates for $w \le n^{1-\delta}$ where $\delta > 0$ is an arbitrary constant, then $t = \Omega\left(\frac{m\log n}{\log s}\right)$.*

4 Direct-sum richness lemma

In this section, we prove a richness lemma for certificates using direct-sum property of data structure problems. Such a lemma was introduced in [17] for cell-probing schemes, which is used to prove some highest known cell-probe lower bounds with *near-linear spaces*.

Consider a vector of problems $\bar{f} = (f_1, \ldots, f_k)$ where every $f_i : X \times Y \to \{0,1\}$ is defined on the same domain $X \times Y$. Let $\bigoplus^k \bar{f} : ([k] \times X) \times Y^k \to \{0,1\}$ be a problem defined as follows: $\bigoplus^k \bar{f}((i,x), \bar{y}) = f_i(x, y_i)$ for every $(i,x) \in [k] \times X$ and every $\bar{y} = (y_1, y_2, \ldots, y_k) \in Y^k$. In particular, for a problem f we denote $\bigoplus^k f = \bigoplus^k \bar{f}$ where $\bar{f}$ is a tuple of k copies of problem f.

Lemma 14 (direct-sum richness lemma for certificates) *Let $\bar{f} = (f_1, f_2 \ldots f_k)$ be a vector of problems such that for each $i = 1, 2, \ldots, k$, we have $f_i : X \times Y \to \{0,1\}$ and f_i is $(\alpha|X|, \beta|Y|)$-rich. If problem $\bigoplus^k \bar{f}$ has (s, w, t)-certificates for a $t \le \frac{s}{k}$, then there exists a $1 \le i \le k$ such that f_i contains a monochromatic 1-rectangle of size $\frac{\alpha^{O(1)}|X|}{2^{O(t\log\frac{s}{kt})}} \times \frac{\beta^{O(1)}|Y|}{2^{O(wt+t\log\frac{s}{kt})}}$.*

Remark 1. The direct-sum richness lemma proved in [17] is for asymmetric communication protocols as well as cell-probing schemes, and gives a rectangle size of $\frac{\alpha^{O(1)}|X|}{2^{O(t\log\frac{s}{k})}} \times \frac{\beta^{O(1)}|Y|}{2^{O(wt+t\log\frac{s}{k})}}$. Our direct-sum richness lemma has a better rectangle bound. This improvement may support stronger lower bounds which separate between linear and near-linear spaces.

Remark 2. A key idea to apply this direct sum based lower bound scheme is to exploit the extra power gained by the model from solving k problem instances in parallel. In [17], this is achieved by seeing cell probes as communications between query algorithm and table, and t-round adaptive cell probes for answering k parallel queries can be expressed in $t\log\binom{s}{k}$ bits instead of naively $kt\log s$ bits. For our direct-sum richness lemma for certificates, in contrast, we will see (in Lemma 15) that unlike communications, the parallel simulation of certificates does not give us any extra gain, however, in our case all extra gains are provided by the improved bound in Lemma 4, the richness lemma for certificates. Indeed, all our extra gains by "parallelism" are offered by the one-line reduction in Lemma 7, which basically says that the certificates for k instances of a problem can be expressed in $\log\binom{s}{kt}$ bits, even better than the $t\log\binom{s}{k}$-bit bound for communications. Giving

up adaptivity is essential to this improvement on the power of parallelism, so that all kt cells can be chosen at once which gives the $\log \binom{s}{kt}$-bit bound: *we are now not even parallel over instances, but also parallel over time.*

The idea of proving Lemma 14 can be concluded as: (1) reducing the problem $\bigoplus^k \bar{f}$ from a direct-product problem $\bigwedge^k \bar{f}$ whose richness and monochromatic rectangles can be easily translated between $\bigwedge^k \bar{f}$ and subproblems f_i; and (2) applying Lemma 4, the richness lemma for certificates, to obtain large monochromatic rectangles for the direct-product problem.

We first define a direct-product operation on vector of problems. For $\bar{f} = (f_1, \ldots, f_k)$ with $f_i : X \times Y \to \{0,1\}$ for every $1 \le i \le k$, let $\bigwedge^k \bar{f} : X^k \times Y^k \to \{0,1\}$ be a direct-product problem defined as: $\bigwedge^k \bar{f}(\bar{x}, \bar{y}) = \prod_i f_i(x_i, y_i)$ for every $\bar{x} = (x_1, \ldots, x_k)$ and every $\bar{y} = (y_1, \ldots, y_k)$.

Lemma 15 *For any $\bar{f} = (f_1, \ldots, f_k)$, if $\bigoplus^k \bar{f}$ has (s, w, t)-certificates for a $t \le \frac{s}{k}$, then $\bigwedge^k \bar{f}$ has (s, w, kt)-certificates.*

Proof: Suppose that $T : Y^k \to \Sigma^s$ with $\Sigma = \{0,1\}^w$ is the code used to encode databases to tables in the (s, w, t)-certificates of $\bigoplus^k \bar{f}$. For problem $\bigwedge^k \bar{f}$, we use the same code T to prepare table. And for each input $(\bar{x}, \bar{y})$ of problem $\bigwedge^k \bar{f}$ where $\bar{x} = (x_1, \ldots, x_k)$ and $\bar{y} = (y_1, \ldots, y_k)$, suppose that for each $1 \le i \le k$, $P_i \subset [s]$ with $|P_i| = t$ is the set of t cells in table $T_{\bar{y}}$ to uniquely identify the value of $\bigoplus^k \bar{f}((i, x_i), \bar{y})$, then let $P = P_1 \cup P_2 \cup \cdots \cup P_k$ so that $|P| \le kt$. It is easy to verify that the set P of at most kt cells in $T_{\bar{y}}$ uniquely identifies the value of $\bigwedge^k \bar{f}(\bar{x}, \bar{y}) = \bigwedge_{1 \le i \le k} \left(\bigoplus^k \bar{f}((i, x_i), \bar{y}) \right)$ because it contains all cells which can uniquely identify the value of $\bigoplus^k \bar{f}((i, x_i), \bar{y})$ for every $1 \le i \le k$. Therefore, problem $\bigwedge^k \bar{f}$ has (s, w, kt)-certificates. ∎

The following two lemmas are from [17]. These lemmas give easy translations of richness and monochromatic rectangles between the direct-product problem $\bigwedge^k \bar{f}$ and subproblems f_i.

Lemma 16 (Pătraşcu and Thorup [17]) *If $\bar{f} = (f_1, f_2 \ldots f_k)$ has $f_i : X \times Y \to \{0,1\}$ and f_i is $(\alpha|X|, \beta|Y|)$-rich for every $1 \le i \le k$, then $\bigwedge^k \bar{f}$ is $((\alpha|X|)^k, (\beta|Y|)^k)$-rich.*

Lemma 17 (Pătraşcu and Thorup [17]) *For any $\bar{f} = (f_1, \ldots, f_k)$ with $f_i : X \times Y \to \{0,1\}$ for every $1 \le i \le k$, if $\bigwedge^k \bar{f}$ contains a monochromatic 1-rectangle of size $(\alpha|X|)^k \times (\beta|Y|)^k$, then there exists a $1 \le i \le k$ such that f_i contains a monochromatic 1-rectangle of size $(\alpha)^3|X| \times (\beta)^3|Y|$.*

The direct-sum richness lemma can be easily proved by combining the above lemmas with the richness lemma for certificates.

Proof: [Proof of Lemma 14] If $\bigoplus^k \bar{f}$ has (s, w, t)-certificates, then by Lemma 15, the direct-product problem $\bigwedge^k \bar{f}$ has (s, w, kt)-certificates. Since every f_i in $\bar{f} = (f_1, f_2, \ldots, f_k)$ is $(\alpha|X|, \beta|Y|)$-rich, by Lemma 16 we have that $\bigwedge^k \bar{f}$ is $((\alpha|X|)^k, (\beta|Y|)^k)$-rich. Applying Lemma 4, the richness lemma for certificates, problem $\bigwedge^k \bar{f}$ has a 1-rectangle of size $\frac{(\alpha|X|)^k}{\binom{s}{kt}} \times \frac{(\beta|Y|)^k}{\binom{s}{kt} 2^{kwt}}$. Then due to Lemma 17, we have a problem f_i who contains a monochromatic 1-rectangle of size $\frac{\alpha^{O(1)}|X|}{2^{O(t \log \frac{s}{kt})}} \times \frac{\beta^{O(1)}|Y|}{2^{O(wt + t \log \frac{s}{kt})}}$. ∎

4.1 Applications

We then apply the direct-sum richness lemma to prove lower bounds for two important high dimensional problems: approximate near neighbor (ANN) in hamming space and partial match (PM).

- For ANN in d-dimensional hamming space, we prove a $t = \Omega(d/\log \frac{sw}{nd})$ lower bound for (s, w, t)-certificates. The highest known cell-probing scheme lower bound for the problem is $t = \Omega(d/\log \frac{sw}{n})$. In a super-linear space, our certificate lower bound matches the highest known lower bound for cell-probing scheme; and for linear space, our lower bound becomes $t = \Omega(d)$, which gives a strict improvement, and also matches the highest cell-probe lower bound ever known for any problem (which has only been achieved for polynomial evaluation [8]).
- For d-dimensional PM, we prove a $t = \Omega(d/\log \frac{sw}{n})$ lower bound for (s, w, t)-certificates, which matches the highest known cell-probing scheme lower bound for the problem in [17].

4.1.1 Approximate near neighbor (ANN)

The near neighbor problem NN_n^d in a d-dimensional metric space is defined as follows: a database y contains n points from a d-dimensional metric space, for any query point x from the same space and a distance threshold λ, the problem asks whether there is a point in database y within distance λ from x. The approximate near neighbor problem $\mathrm{ANN}_n^{\lambda,\gamma,d}$ is similarly defined, except upon a query x to a database y, answering "yes" if there is a point in database y within distance λ from x and "no" if all points in y are $\gamma\lambda$-far away from x (and answering arbitrarily if otherwise).

We first prove a lower bound for $\mathrm{ANN}_n^{\lambda,\gamma,d}$ in Hamming space $X = \{0,1\}^d$, where for any two points $x, x' \in X$ the distance between them is given by Hamming distance $h(x, x')$.

The richness and monochromatic rectangles of $\mathrm{ANN}_n^{\lambda,\gamma,n}$ were analyzed in [9].

Claim 18 (Claim 10 and 11 in [9]) *There is a $\lambda \le d$ such that $\mathrm{ANN}_n^{\lambda,\gamma,d}$ is $(2^{d-1}, 2^{nd})$-rich and $\mathrm{ANN}_n^{\lambda,\gamma,d}$ does not contain a 1-rectangle of size $2^{d-d/(169\gamma^2)} \times 2^{nd-nd/(32\gamma^2)}$.*

A model-independent self-reduction of ANN was constructed in [17].

Claim 19 (Theorem 6 in [17]) *For $D = d/(1+5\gamma) \ge \log n$, $N < n$ and $k = n/N$, there exist two functions ϕ_X, ϕ_Y such that ϕ_X (and ϕ_Y) maps each query (x, i) (and database $\bar{y}$) of $\bigoplus^k \mathrm{ANN}_N^{\lambda,\gamma,D}$ to a query x' (and database y') of $\mathrm{ANN}_n^{\lambda,\gamma,d}$ and it holds that $\bigoplus^k \mathrm{ANN}_N^{\lambda,\gamma,D}((x, i), \bar{y}) = \mathrm{ANN}_n^{\lambda,\gamma,d}(x', y')$.*

We then prove the following certificate lower bound for ANN.

Theorem 20 *For $\mathrm{ANN}_n^{\lambda,\gamma,d}$ in d-dimensional Hamming space, assuming $d \ge (1+5\gamma)\log n$, there exists a λ, such that if $\mathrm{ANN}_n^{\lambda,\gamma,d}$ has (s, w, t)-certificates, then $t = \Omega\left(\frac{d}{\gamma^3}/\log \frac{sw\gamma^3}{nd}\right)$.*

Proof: Due to the model-independent reduction from $\bigoplus^k \mathrm{ANN}_N^{\lambda,\gamma,D}$ to $\mathrm{ANN}_n^{\lambda,\gamma,d}$ of Claim 19, existence of (s, w, t)-certificates for $\mathrm{ANN}_n^{\lambda,\gamma,d}$ implies the existence of (s, w, t)-certificates for $\bigoplus^k \mathrm{ANN}_N^{\lambda,\gamma,D}$.

Note that for problem $\mathrm{ANN}_N^{\lambda,\gamma,D}$, the size of query domain is $|X| = 2^D$, and the size of data domain is $|Y| = 2^{ND}$, so applying Claim 18, the problem is $(|X|/2, |Y|)$-rich. Assuming that $t \le \frac{s}{k}$, by Lemma 14, $\mathrm{ANN}_N^{\lambda,\gamma,D}$ contains a 1-rectangle of size $2^D/2^{O(t\log\frac{s}{kt})} \times 2^{ND}/2^{O(wt+t\log\frac{s}{kt})}$. Due to Claim 18, and by a calculation, we have either $t = \Omega\left(\frac{D}{\gamma^2}/\log\frac{s}{kt}\right)$ or $t = \Omega\left(\frac{ND}{\gamma^2}/w\right)$. We

then choose $N = w$. Note that such choice of N may violate the assumption $t \leq \frac{s}{k}$ (that is, $N \geq \frac{tn}{s}$) only when it implies an even higher lower bound $t > \frac{sw}{n}$. With this choice of $N = w$, the bound $t = \Omega\left(\frac{D}{\gamma^2}/\log\frac{s}{kt}\right)$ is the smaller one in the two branches. Substituting $D = d/(1+5\gamma)$ and $k = n/N$ we have $t = \Omega\left(\frac{d}{\gamma^3}/\log\frac{sN}{nt}\right) = \Omega\left(\frac{d}{\gamma^3}/\log\frac{sw}{nt}\right)$. Multiplying both side by a $\Delta = \frac{sw}{nd}$ gives us $\Delta\cdot\gamma^3 = \Omega\left(\frac{\Delta d}{t}/\log\frac{\Delta d}{t}\right)$. Assuming $\Delta' = \frac{\Delta d}{t}$, we have $\frac{\Delta'}{\log\Delta'} = O(\Delta\gamma^3)$. The function $f(x) = \frac{x}{\log x}$ is increasing for $x > 1$, so we have $\Delta' = O(\Delta\gamma^3\log(\Delta\gamma^3))$, which gives us the lower bound $t = \Omega\left(\frac{d}{\gamma^3}/\log\frac{sw\gamma^3}{nd}\right)$. ∎

For general space, when points are still from the Hamming cube $\{0,1\}^d$, for any two points $x, x' \in \{0,1\}^d$, the Hamming distance $h(x,x') = \|x - x'\|_1 = \|x - x'\|_2^2$. And by setting $\gamma = 1$, we have the following corollary for exact near neighbor.

Corollary 21 *There exists a constant C such that for problem NN_n^d with Hamming distance, Manhattan norm ℓ_1 or Euclidean norm ℓ_2, assuming $d \geq C\log n$, if NN_n^d has (s,w,t)-certificates, then $t = \Omega(d/\log\frac{sw}{nd})$.*

4.1.2 Partial match

The partial match problem is another fundamental high-dimensional problem. The d-dimensional partial match problem PM_n^d is defined as follows: a database y contains n strings from $\{0,1\}^d$, for any query pattern $x \in \{0,1,*\}^d$, the problem asks whether there is a string z in database y matching pattern x, in such a way that $x_i = z_i$ for all $i \in [d]$ that $x_i \neq *$.

Theorem 22 *Assuming $d \geq 2\log n$, if problem PM_n^d has (s,w,t)-certificates for a $w = d^{O(1)}$, then $t = \Omega\left(d/\log\frac{sd}{n}\right)$.*

Proof: The proof is almost exactly the same as the proof of partial match lower bound in [17]. We restate the proof in the context of certificates. Let $N = n/k$ and $D = d - \log k \geq d/2$. We have the following model-independent reduction from $\bigoplus^k \mathrm{PM}_N^D$ to PM_n^d: For the data input $\bigoplus^k \mathrm{PM}_N^D$, we add the subproblem index in binary code, which takes $\log k$ bits, as a prefix for every string. And for the query, we also add the subproblem index i in binary code as a prefix to the query pattern to form a new query in PM_n^d. It is easy to see PM_n^d solves $\bigoplus^k \mathrm{PM}_N^D$ with such a reduction, and (s,w,t)-certificates for PM_n^d are (s,w,t)-certificates for $\bigoplus^k \mathrm{PM}_N^D$.

In Theorem 11 of [17], it is proved that on a certain domain $X \times Y$ for PM_N^D:

- PM_N^D is $(|X|/4, |Y|/4)$-rich. In fact, in [17] it is only proved that the density of 1s in PM_N^D is at least $1/2$, which easily implies the richness due to an averaging argument.
- PM_N^D has no 1-rectangle of size $|X|/2^{O(D)} \times |Y|/2^{O(\sqrt{N}/D^2)}$.

Assuming that $t \leq \frac{s}{k}$, by Lemma 14, we have either $t\log\frac{s}{k} = \Omega(D)$ or $t\log\frac{s}{k} + wt = \Omega(\sqrt{N}/D^2)$. We choose $N = w^2\cdot D^8$. Note that this choice of N may violate the assumption $t \leq \frac{s}{k}$ only when an even higher lower bound $t > \frac{sw^2D^8}{n} = \Omega(d^2)$ holds. With this choice of $N = w^2\cdot D^8 = d^{O(1)}$, the second bound above becomes $t = \Omega(d^2)$, while the first becomes $t = \Omega\left(d/\log\frac{sd}{nt}\right) = \Omega\left(d/\log\frac{sd}{n}\right)$. ∎

It is well known that partial match can be reduced to 3-approximate near neighbor in ℓ_∞-norm by a very simple reduction [6]. We write 3-$\mathrm{ANN}_n^{\lambda,d}$ for $\mathrm{ANN}_n^{\lambda,3,d}$.

Theorem 23 *Assuming $d \geq 2\log n$, there is a λ such that if 3-ANN$_n^{\lambda,d}$ in ℓ_∞-norm has (s,w,t)-certificates for a $w = d^{O(1)}$, then $t = \Omega(d/\log \frac{sd}{n})$.*

Proof: We have the following model-independent reduction. For each query pattern x of partial match, we make the following transformation to each coordinate: $0 \to -\frac{1}{2}$; $* \to \frac{1}{2}$; $1 \to \frac{3}{2}$. For a string in database the ℓ_∞-distance is $\frac{1}{2}$ if it matches pattern x and $\frac{3}{2}$ if otherwise. ∎

5 Lower bounds implied by lopsided set disjointness

It is observed in [15] that a variety of cell-probe lower bounds can be deduced from the communication complexity of one problem, the lopsided set disjointness (LSD). In [18], the communication complexity of LSD is also used to prove the cell-probe lower bound for approximate distance oracle.

In this section, we modify these communication-based reductions to make them model-independent. A consequence of this is a list of certificate lower bounds which match the highest known cell-probe lower bounds for respective problems, including: 2-Blocked-LSD, reachability oracle, 2D stabbing, 2D range counting, 4D range reporting, and approximate distance oracle.

5.1 LSD with structures

A key idea of using LSD in reduction is to reduce from LSD with restricted inputs.

For the purpose of reduction, the LSD problem is usually formulated as follows: the universe is $[N \cdot B]$, each query set $S \subset [N \cdot B]$ has size N, and there is no restriction on the size of data set $T \subseteq [N \cdot B]$. The LSD problem asks whether S and T are disjoint.

Proposition 24 *For any $M \geq N$, if LSD has monochromatic 1-rectangle of size $\binom{M}{N} \times K$ then $K \leq 2^{NB-M}$.*

Proof: For any 1-rectangle of LSD, suppose the rows are indexed by $S_1, S_2, \ldots, S_R$ and the columns are indexed by $T_1, T_2, \ldots, T_K$. Consider the set $\mathcal{S} = \bigcup_i S_i$. Let $M = |\mathcal{S}|$. Note that $R \leq \binom{M}{N}$. For any T_i, we have $T_i \cap \mathcal{S} = \emptyset$, so it holds that $K \leq 2^{NB-M}$. ∎

The 2-Blocked-LSD is a special case of LSD problem: the universe $[N \cdot B]$ is interpreted as $[\frac{N}{B}] \times [B] \times [B]$ and it is guaranteed that for every $x \in [\frac{N}{B}]$ and $y \in [B]$, S contains a single element of the form $(x, y, *)$ and a single element of the form $(x, *, y)$.

In [15], general LSD problem is reduced to 2-Blocked-LSD by communication protocols. Here we translate this reduction in the communication model to a model-independent reduction from subproblems of LSD to 2-Blocked-LSD.

The following claim can be proved by a standard application of the probabilistic method.

Claim 25 (Lemma 11 in [14]) *There exists a set $\mathcal{F}$ of permutations on universe $[N \cdot B]$, where $|\mathcal{F}| = e^{2N} \cdot 2N \log B$, such that for any query set $S \subset [N \cdot B]$ of LSD, there exists a permutation $\pi \in \mathcal{F}$ for which $\pi(S)$ is an instance of 2-Blocked-LSD.*

We then state our model-independent reduction as the following certificate lower bound.

Theorem 26 *For any constant $\delta > 0$, if 2-Blocked-LSD on universe $[\frac{N}{B}] \times [B] \times [B]$ has (s,w,t)-certificates, then it holds either $t = \Omega\left(\frac{NB^{1-\delta}}{w}\right)$ or $t = \Omega\left(\frac{N \log B}{\log \frac{s}{t}}\right)$.*

Proof: By Claim 25, we know there exists a small set $\mathcal{F}$ of permutations for the universe $[N \cdot B]$ such that $|\mathcal{F}| = 2^{O(N)}$ and for any input S of LSD, there exists $\pi \in \mathcal{F}$ for which $\pi(S)$ is an instance of 2-Blocked-LSD. By averaging principle, there exists a $\pi \in \mathcal{F}$ such that for at least $|X|/2^{O(N)}$ many sets S, $\pi(S)$ is an instance of 2-Blocked-LSD. Denote the set of these S as $\mathcal{X}$. Restrict LSD to the domain $\mathcal{X} \times Y$ and denote this subproblem as $\text{LSD}_{\mathcal{X}}$. Obviously $\text{LSD}_{\mathcal{X}}$ can be solved by 2-Blocked-LSD by transforming the input with permutation π, and hence $\text{LSD}_{\mathcal{X}}$ has (s, w, t)-certificates. For any $S \in \mathcal{X}$, there are 2^{NB-N} choices of $T \in Y$ such that $S \cap T = \emptyset$, so the density of 1 in $\text{LSD}_{\mathcal{X}}$ is at least $\frac{1}{2^N}$, thus by a standard averaging argument $\text{LSD}_{\mathcal{X}}$ is $(\frac{1}{2^{O(N)}}|\mathcal{X}|, \frac{1}{2^{O(N)}}|Y|)$-rich. Now by the richness lemma, there exists a $|X|/2^{O(N+t\log\frac{s}{t})} \times |Y|/2^{O(N+t\log\frac{s}{t}+wt)}$ 1-rectangle of $\text{LSD}_{\mathcal{X}}$, which is certainly a 1-rectangle of LSD. Due to Proposition 24, for any $M \geq N$, LSD has no 1-rectangle of size greater than $\binom{M}{N} \times 2^{NB-M}$, which gives us either $N + t\log\frac{s}{t} = \Omega(N\log B - N\log\frac{M}{N})$ or $N + tw + t\log\frac{s}{t} = \Omega(M)$. By setting $M = NB^{1-\delta}$, we prove the theorem. ∎

5.2 Reachability oracle

The problem of reachability oracle is defined as follows: a database stores a (sparse) directed graph G, and reachability queries (can u be reached from v in G?) are answered. The problem is trivially solved, even in the sense of certificates, in quadratic space by storing answers for all pairs of vertices. Solving this problem using near-linear space appears to be very hard. This is proved in [15] for communication protocols as well as for cell-probing schemes. We show the method in [15] can imply the same lower bound for data structure certificates.

Theorem 27 *If reachability oracle of n-vertices graphs has (s, w, t)-certificates for $s = \Omega(n)$, then $t = \Omega\left(\log n/\log\frac{sw}{n}\right)$.*

The lower bound is proved for a special class of graphs, namely butterfly graphs. Besides implying the general reachability oracle lower bound, the special structure of butterfly graphs is very convenient for reductions to other problems.

A butterfly graph is defined by degree b and depth d. The graph has $d+1$ layers, each having b^d vertices. The vertices on level 0 are source vertices with 0 in-degree and the the ones on level d are sinks with 0 out-degree. On each level, each vertex can be regarded as a vector in $[b]^d$. For each non-sink vector (vertex) on level i, there is an edge connecting a vector (vertex) on the $(i+1)$-th level that may differ only on the i-th coordinate. Therefore each non-sink vertex has out-degree b.

The problem $\text{Butterfly-RO}_{n,b}$ is the reachability oracle problem defined on subgraphs of the butterfly graph uniquely specified by degree b and number of non-sink vertices n. For a problem $f : X \times Y \to \{0,1\}$ we define $\bigotimes^k f : X^k \times Y \to \{0,1\}$ as that $\bigotimes^k f(\bar{x}, y) = \prod_{i=1}^k f(x_i, y)$ for any $\bar{x} = (x_1, x_2, \ldots, x_k) \in X^k$ and any $y \in Y$. We further specify that in reachability oracle problem, the answer is a bit indicating the reachability, thus $\bigotimes^k \text{Butterfly-RO}_{n,b}$ is well-defined.

It is discovered in [15] a model-independent reduction from 2-Blocked-LSD on universe $[\frac{N}{B}] \times [B] \times [B]$ to $\bigotimes^k \text{Butterfly-RO}_{N,B}$ for $k = \frac{N}{d}$, where $d = \Theta(\frac{\log N}{\log B})$ is the depth of the butterfly graph. This can be used to prove the following certificate lower bound

Lemma 28 *If $\text{Butterfly-RO}_{N,B}$ has (s, w, t)-certificates, then either $t = \Omega\left(\frac{d\sqrt{B}}{w}\right)$, or $t = \Omega\left(\frac{d\log B}{\log\frac{sd}{N}}\right)$, or $t = \Omega\left(\frac{ds}{N}\right)$, where $d = \Theta\left(\frac{\log N}{\log B}\right)$ is the depth of the butterfly graph.*

Proof: By the same way of straightforwardly combining certificates as in the proof of Lemma 15, assuming that $\frac{N}{d}t \leq s$, if Butterfly-RO$_{N,B}$ has (s, w, t)-certificates then $\bigotimes^k$ Butterfly-RO$_{N,B}$ with $k = \frac{N}{d}$ has $(s, w, \frac{N}{d}t)$-certificates. Violating the assumption of $\frac{N}{d}t \leq s$ gives us $t = \Omega\left(\frac{ds}{N}\right)$. By the model-independent reduction in [15], 2-Blocked-LSD on universe $[\frac{N}{B}] \times [B] \times [B]$ has $(s, w, \frac{N}{d}t)$-certificates. Due to Theorem 26, for any constant $\delta > 0$, either $\frac{N}{d}t = \Omega\left(\frac{NB^{1-\delta}}{w}\right)$ or $\frac{N}{d}t = \Omega\left(\frac{N \log B}{\log \frac{sd}{Nt}}\right) = \Omega\left(\frac{N \log B}{\log \frac{sd}{N}}\right)$. By setting $\delta = \frac{1}{2}$, we have either $t = \Omega(\frac{d\sqrt{B}}{w})$ or $t = \Omega\left(\frac{d \log B}{\log \frac{sd}{N}}\right)$. ∎

Theorem 27 for general graphs is an easy consequence of this lemma.

Proof: [Proof of Theorem 27] Suppose the input graphs are just those of Butterfly-RO$_{N,B}$. By Lemma 28, either $t = \Omega(\frac{d \log B}{\log \frac{sd}{N}})$, or $t = \Omega(\frac{d\sqrt{B}}{w})$, or $t = \Omega(\frac{ds}{N})$. Assuming $s = \Omega(n)$, the third branch becomes $t = \Omega(d)$. Choose B to satisfy $\log B = \max\{2 \log w, \log \frac{sd}{N}\} = \Theta(\log \frac{sdw}{N})$. Then we have $t = \Omega(d)$ for the first and second branches. Since $d = \Theta(\log N / \log B)$, we have $t = \Omega(\log N / \log \frac{sdw}{N}) = \Omega(\log n / \log \frac{sw}{n})$. ∎

Applying the model-independent reductions introduced in [15] from Butterfly-RO$_{n,b}$ to 2D stabbing, 2D range counting, and 4D range reporting, we have the certificate lower bounds which match the highest known lower bounds for cell-probing schemes for these problems.

Theorem 29 *If 2D stabbing over m rectangles has (s, w, t)-certificates, then $t = \Omega(\log m / \log \frac{sw}{m})$.*

Theorem 30 *If 2D range counting has (s, w, t)-certificates, then $t = \Omega(\log n / \log \frac{sw}{n})$.*

Theorem 31 *If 4D range reporting has (s, w, t)-certificates, then $t = \Omega(\log n / \log \frac{sw}{n})$.*

5.3 Approximate distance oracle

For the distance oracle problem, distance queries $d_G(u, v)$ are answered for a database graph G. For this fundamental problem, approximation is very important because exact solution appears to be very difficult for nontrivial settings. Given a stretch factor $\alpha > 1$, the α-approximate distance oracle problem can be defined as: for each queried vertex pair (u, v) and a distance threshold $\tilde{d}$, the problem is required to distinguish between the two cases $d_G(u, v) \leq \tilde{d}$ and $d_G(u, v) \geq \alpha\tilde{d}$.

We prove the following certificate lower bound for approximate distance oracle which matches the lower bound proved in [18] for cell-probing schemes.

Theorem 32 *If α-approximate distance oracle has (s, w, t)-certificates, then $t = \Omega\left(\frac{\log n}{\alpha \log(s \log n/n)}\right)$. This holds even when the problem is restricted to sparse graphs with max degree $\mathrm{poly}(tw\alpha / \log n)$ for an $\alpha = o\left(\frac{\log n}{\log(w \log n)}\right)$.*

We use the following notations introduced in [18]. For graph $G = (V, E)$ and any two positive integers k, ℓ, let $\mathcal{P}(G, \ell, k)$ be the set whose elements are all possible sets $P \subseteq E$ where P can be written as a union of k vertex-disjoint paths in G, each of length exactly ℓ. Let $g(G)$ denote the girth of graph G. The following claim, which is quite similar to Claim 25, is proved in [18] by the same probabilistic argument.

Claim 33 (Claim 13 in [18]) *Let $k, \ell > 0$ be two integers and $N = k\ell$. Let $G = (V, E)$ be a graph with $|E| = B \cdot N$ for a positive integer B, and $\mathcal{P} = \mathcal{P}(G, \ell, k)$. There exist m bijections $f_1, \ldots, f_m : [NB] \to E$, where $m = \ln((\mathrm{e}B)^N) \cdot \frac{(\mathrm{e}B)^N}{|\mathcal{P}|}$, such that for any $S \subseteq [NB]$ with $|S| = N$, there is a bijection f_i such that $f_i(S) \in \mathcal{P}(G, \ell, k)$.*

Consider the problem of α-approximate distance oracle for base-graph G, in which the α-approximate distance queries are answered only for spanning subgraphs of G. The following lemma is the certificate version of a key theorem in [18].

Lemma 34 *There exists a universal constant C such that the following holds. Let $G = (V, E)$ be a graph, such that α-approximate distance oracle for the base-graph G has (s, w, t)-certificates. Let k, ℓ be two positive integers, such that $\ell < \frac{g(G)}{\alpha+1}$. Assume $|E| \geq k\ell(2tw/\ell)^{1/C}$. Then*

$$s \geq \frac{k}{e} \left(\frac{|\mathcal{P}(G, \ell, k)|^{1/k\ell}}{e(|E|/k\ell)^{1-C}} \right)^{\frac{\ell}{t}} (e|E|)^{-\frac{1}{tk}}$$

Proof: Suppose $N = k\ell$ and $B = |E|/N$. Consider the LSD problem LSD: $X \times Y \to \{0, 1\}$ defined on universe $[N \cdot B]$ such that each query set $S \subset [N \cdot B]$ is of size $|S| = N$ and each dataset $T \subseteq [N \cdot B]$ is of arbitrary size. By Claim 33, there exists m bijections, $f_1, \ldots, f_m : [NB] \to E$, where $m = \ln((eB)^N) \cdot \frac{(eB)^N}{|\mathcal{P}|}$ where $\mathcal{P} = \mathcal{P}(G, \ell, k)$, such that for any $S \subseteq [NB]$ with $|S| = N$, there exists a bijection f_i such that $f_i(S) \in \mathcal{P}(G, \ell, k)$. By averaging principle, there exists an f_i such that for at least $|X|/m$ many sets S, it holds that $f_i(S) \in \mathcal{P}(G, l, k)$. Denote the set of such S as $\mathcal{X}$. Restrict LSD to the domain $\mathcal{X} \times Y$ and denote this subproblem as $\mathrm{LSD}_{\mathcal{X}}$. Next we prove $\mathrm{LSD}_{\mathcal{X}}$ can be solved by a composition of α-approximate distance oracles.

Let f_i be the bijection such that $f_i(S) \in \mathcal{P}(G, \ell, k)$ for all $S \in \mathcal{X}$. For any $S \in \mathcal{X}, T \subseteq [N \cdot B]$, an instance for approximate distance oracle for the base graph $G = (V, E)$ is constructed as follows. The database graph for distance oracle is the spanning subgraph $G' = (V, E')$ where $E' = E \setminus f_i(T)$. Due to the property of bijection f_i, it holds that $P = f_i(S)$ contains k vertex-disjoint paths $p_1, p_2, \ldots, p_k$, each of length ℓ. Let $(u_1, v_1), \ldots, (u_k, v_k)$ denote the pairs of end-vertices of these paths. Since f_i is a bijection, the disjointness of S and T translates to the disjointness of $f_i(S)$ and $f_i(T)$, i.e. all these k vertex-disjoint paths are intact by removing edges in $f_i(T)$ from the graph G.

Consider the α-approximate distance oracle problem α-Dist_G for the base-graph G. We then observe that $\mathrm{LSD}_{\mathcal{X}}$ can be solved by the problem $\bigotimes^k \alpha$-Dist_G of answering k parallel approximate distance queries, where $\bigotimes^k f$ of a problem f is as defined in last section. Consider the k vertex pairs $(u_i, v_i), i = 1, 2, \ldots, k$ connected by vertex-disjoint paths p_i constructed above. We have $d_G(u_i, v_i) = \ell$ for every $1 \leq i \leq k$. For α-Dist_G, if all edges in p_i are in E', then $d_{G'}(u_i, v_i) \leq \ell$, so α-$\mathrm{Dist}_G((u_i, v_i, \ell), G')$ will return "yes", and if there is an edge in p_i is not in E', since graph G has girth $g(G) > (\alpha + 1)\ell$, we must have $d_{G'}(u_i, v_i) \geq g(G) - \ell > \alpha\ell$, so α-$\mathrm{Dist}_G((u_i, v_i, \ell), G')$ will return "no". By above discussion, if α-$\mathrm{Dist}_G((u_i, v_i, \ell), G')$ returns "yes" for all k queries then it must hold $S \cap T = \emptyset$, and if α-$\mathrm{Dist}_G((u_i, v_i, \ell), G')$ returns "no" for some i, then $S \cap T \neq \emptyset$, i.e. we have a model-independent reduction from $\mathrm{LSD}_{\mathcal{X}}$ to $\bigotimes^k \alpha$-Dist_G.

If the α-approximate distance oracle problem α-Dist_G has (s, w, t)-certificates, then by directly combining k certificates for k parallel queries, the problem $\bigotimes^k \alpha$-Dist_G has (s, w, kt)-certificates, and hence $\mathrm{LSD}_{\mathcal{X}}$ has (s, w, kt)-certificates. For every $S \in \mathcal{X}$, there are 2^{NB-N} many T disjoint with S, so the density of $\mathrm{LSD}_{\mathcal{X}}$ is at least 2^{-N}. By a standard averaging argument, this means

$\mathrm{LSD}_{\mathcal{X}}$ is $(\frac{1}{2^{N+1}}|\mathcal{X}|, \frac{1}{2^{N+1}}|Y|)$-rich. By Lemma 4, there exist universal constants $C_1, C_2 > 0$ such that $\mathrm{LSD}_{\mathcal{X}}$ has monochromatic 1-rectangle of size $|\mathcal{X}|/2^{O(N+kt\log\frac{s}{kt})} \times |Y|/2^{O(N+kt\log\frac{s}{kt}+ktw)}$, which is also 1-rectangle of LSD. Note that $|\mathcal{X}| \geq |X|/m = |X|/\ln((eB)^N) \cdot \frac{(eB)^N}{|\mathcal{P}|}$, so the rectangle is of size at least

$$|X|/2^{O(N\log(eB)+\log(eBN)-\log(|\mathcal{P}|)+kt\log\frac{s}{kt})} \times |Y|/2^{O(N+kt\log\frac{s}{kt}+ktw)},$$

where the big-O notations hide only universal constants. And for LSD, $|X| = \binom{NB}{N}$ and $|Y| = 2^{NB}$. Due to Proposition 24, for any $M \geq N$, LSD has no 1-rectangle of size greater than $\binom{M}{N} \times 2^{NB-M}$. By a calculation, there exist a universal constant $C > 0$ such that by considering an $M = \Theta(NB^C)$, we have either $tk\log(s/k) + N\log(eB) + \log(eBN) - \log(|\mathcal{P}|) \geq CN\log B$ or $ktw \geq NB^C$. Since the lemma assumes $|E| \geq k\ell(2tw/\ell)^{1/C}$, we have $B \geq (2tw/\ell)^{1/C}$, thus $NB^C \geq k\ell \cdot 2tw/\ell = 2ktw$. The second branch can never be satisfied. And by a calculation, the first branch gives us the bound of the lemma. ∎

The following graph-theoretical theorem is proved in [18].

Theorem 35 (combining Lemma 14, Theorem 9, 17, and 18 of [18]) *Let n be sufficiently large. For any constant $C > 0$, any $t = t(n)$ and any $\alpha = \alpha(n), w = w(n)$ satisfying $w = n^{o(1)}$ and $\alpha = o\left(\frac{\log n}{\log(w\log n)}\right)$. There exist $r = r(n)$ and r-regular graph $G = G_n$ of n vertices, such that*

- $r \geq (4tw\alpha/g(G))^{1/C}$;
- $2\alpha \leq g(G) \leq \log n$;
- $|\mathcal{P}(G,\ell,k)|^{1/k\ell} = \Omega(r)$ *for* $\ell = \lfloor g(G)/2\alpha \rfloor$ *and* $k = n/20\ell$;
- $r^{g(G)} = n^{\Omega(1)}$.

Now we prove Theorem 32 by applying Lemma 34 to the sequence of regular graphs G_n constructed in Theorem 35. Note that in G_n, we have $|E| = n\cdot r/2 = 10k\ell\cdot r \geq 10k\ell\cdot(2tw/\ell)^{1/C} \geq k\ell(2tw/\ell)^{1/C}$, so the assumption of Lemma 34 is satisfied. On the other hand, we have $|\mathcal{P}(G,\ell,k)|^{1/k\ell} = \Omega(r)$ and $e(|E|/k\ell)^{1-C} = \Theta(r^{1-C})$. Since $|E| \leq n^2 \leq (\ell k)^4 \leq k^8$, we have $(e|E|)^{1/tk} = \Theta(1)$. And it holds that $k = \frac{n}{20\ell} = \Omega(\frac{n}{\log n})$. Ignoring constant factors, the bound in Lemma 34 implies:

$$s \geq \frac{n}{\log n}\left(\frac{r}{r^{1-C}}\right)^{\Omega(\ell/t)} = \frac{n}{\log n}r^{\Omega(\ell/t)} = \frac{n}{\log n}r^{\Omega(g(G)/\alpha t)} = \frac{n^{\Omega(1+1/\alpha t)}}{\log n}$$

Translating this to a lower bound of t, we have $t = \Omega\left(\frac{\log n}{\alpha\log(s\log n/n)}\right)$.

Acknowledgment. We are deeply grateful to Kasper Green Larsen for helpful discussions about lower bound techniques in cell-probe model.

References

[1] A. Andoni, P. Indyk, and M. Pătraşcu. On the optimality of the dimensionality reduction method. In *Proc. 47th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 449–458, 2006.

[2] O. Barkol and Y. Rabani. Tighter lower bounds for nearest neighbor search and related problems in the cell probe model. *Journal of Computer and System Sciences*, 64(4):873–896, 2002.

[3] A. Borodin, R. Ostrovsky, and Y. Rabani. Lower bounds for high dimensional nearest neighbor search and related problems. In *Proc. 31st ACM Symposium on Theory of Computing (STOC)*, pages 312–321, 1999.

[4] H. Buhrman and R. De Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[5] T. Husfeldt and T. Rauhe. Hardness results for dynamic problems by extensions of fredman and saks' chronogram method. In *Proc. 25th International Colloquium on Automata, Languages and Programming*, pages 67–78, 1998.

[6] P. Indyk. On approximate nearest neighbors under ℓ_∞ norm. *Journal of Computer and System Sciences*, 63(4):627–638, 2001.

[7] T. Jayram, S. Khot, R. Kumar, and Y. Rabani. Cell-probe lower bounds for the partial match problem. In *Proc. 35th ACM Symposium on Theory of Computing (STOC)*, pages 667–672, 2003.

[8] K. G. Larsen. Higher cell probe lower bounds for evaluating polynomials. In *Proc. 53rd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 293–301, 2012.

[9] D. Liu. A strong lower bound for approximate nearest neighbor searching. *Information Processing Letters*, 92(1):23–29, 2004.

[10] P. B. Miltersen, N. Nisan, S. Safra, and A. Wigderson. On data structures and asymmetric communication complexity. *Journal of Computer and System Sciences*, 57(1):37–49, 1998.

[11] R. Panigrahy, K. Talwar, and U. Wieder. A geometric approach to lower bounds for approximate near-neighbor search and partial match. In *Proc. 49th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 414–423, 2008.

[12] R. Panigrahy, K. Talwar, and U. Wieder. Lower bounds on near neighbor search via metric expansion. In *Proc. 51th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 805–814, 2010.

[13] M. Pătraşcu. Lower bounds for 2-dimensional range counting. In *Proc. 30th ACM Symposium on Theory of Computing (STOC)*, pages 40–46, 2007.

[14] M. Pătraşcu. (data) structures. In *Proc. 49th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 434–443, 2008.

[15] M. Pătraşcu. Unifying the landscape of cell-probe lower bounds. *SIAM Journal on Computing*, 40(3):827–847, 2011. See also FOCS'08.

[16] M. Pătraşcu and M. Thorup. Time-space trade-offs for predecessor search. In *Proc. 38th ACM Symposium on Theory of Computing (STOC)*, pages 232–240, 2006.

[17] M. Pătraşcu and M. Thorup. Higher lower bounds for near-neighbor and further rich problems. *SIAM Journal on Computing*, 39(2):730–741, 2010. See also FOCS'06.

[18] C. Sommer, E. Verbin, and W. Yu. Distance oracles for sparse graphs. In *Proc. 50th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 703–712, 2009.

[19] Y. Yin. Cell-probe proofs. *ACM Transactions on Computation Theory (TOCT)*, 2(1):1, 2010.